Afstudeerverslag

DATA TRANSMISSION ACROSS THE FM CHANNEL

J.P.J. Kemmelings
Data transmission across the FM channel
ABSTRACT

When existing Frequency Modulation (FM) channels are used to send digital information, it is important to pay attention to the power spectral density (PSD) of the FM signal to be transmitted. Without this attention the PSD is very likely to become too broad. As a result the FM signal causes interference in neighbour channels (Co Channel Interference, CCI). A modulation technique based on digital FM, which generates an FM signal with a relatively small PSD, is Continuous Phase Modulation (CPM). In the CPM scheme, the baseband pulse train is, prior to transmission, filtered by a Low Pass Filter (LPF). The LPF removes the high frequencies from the baseband signal. As a result the PSD of the FM signal is smaller. When, a Gaussian filter is used as the implementation of the LPF, the CPM system is referred to as Gaussian Minimum Shift Keying (GMSK). It has been proven in literature that with GMSK good results can be obtained. Coding of the data prior to transmission has also its influence on the PSD of the generated FM signal. A largely used group of codes are the so called Run Length Limited (RLL) codes. These codes have the property that the amount of information per symbol is smaller than 1 bit. It is also possible to increase the information per symbol by using more levels. In this work, binary RLL coded GMSK schemes, 4-level GMSK schemes and uncoded GMSK schemes are compared.

Filtering, coding and the symbol rate influence the amount of information that can be transmitted. The PSD of the FM signal is the high frequency side of the problem, on the other hand there is the baseband behaviour. The question is how to compare the different schemes. The baseband behaviour is judged by comparing eye-patterns. The high frequency behaviour is judged by looking at the PSD of the FM signals.

The conclusion of this work is that 4-level GMSK, judged on the properties mentioned above, clearly outperforms RLL coded GMSK. The performance of uncoded GMSK is better than RLL coded GMSK but worse than 4-level GMSK.

The first assignment was to examine a practical FM system. This was a separate assignment, it was not used to answer the real assignment of this work (which is the comparison of the schemes mentioned above). Using echo cancellation techniques, a model of the baseband connection of the FM system is determined. Important aspects are the presence of Inter Symbol Interference (ISI), nonlinearities and the shape of the baseband noise PSD. The conclusion found is that the baseband behaviour of the examined FM system differs significantly from the theoretical baseband behaviour of an FM system.

The investigation showed that 4-level GMSK is the better one. For further investigations it is interesting to look at 8- or 16-level signaling.
Keywords: GMSK
Premodulation coding
PSD
Contents

1 Introduction .................................................. 7

2 Description of the transmission system ................... 10
   2.1 Introduction .............................................. 10
   2.2 The baseband channel ................................... 11
       2.2.1 Theoretical discussion of the baseband channel of an FM transmission system ..................................... 11
   2.3 Canceller-based channel characterization ............... 13
       2.3.1 Introduction .......................................... 13
       2.3.2 Canceller structures ................................ 16
       2.3.3 LMS-based channel characterization ................ 20
       2.3.4 Results .............................................. 23
   2.4 Continuous Phase Modulation .............................. 27
   2.5 Premodulation coding .................................... 30
       2.5.1 Introduction .......................................... 30
       2.5.2 Run-length-limited coding ............................ 30
       2.5.3 2-Binary-to-1-Quaternary coding .................... 35

3 Comparing the different schemes ......................... 37
   3.1 Introduction .............................................. 37
   3.2 The eye pattern ......................................... 39
   3.3 HF spectrum calculation .................................. 40

© Philips Electronics N.V. 3
4 Results

5 Conclusion

A Characterization results, file 1

A.1 Impulse response ........................................... 57
A.2 Noise PSD .................................................. 58
A.3 Amplitude response ........................................ 59

B Gaussian frequency pulses .................................. 60

C RLL data ..................................................... 61

D Autocorrelation function method .......................... 65

D.1 Introduction ................................................ 65
D.2 The encoder ................................................. 65
D.3 The CPM system ........................................... 67
D.4 Relation between the PSD of the transmitted signal and the PSD of the complex-valued signal ........................................... 68
D.5 The autocorrelation function method .................... 71

E Eye-patterns and PSDs ........................................ 83

E.1 Eye-patterns (1,3) RLL GMSK ............................. 84
E.2 Eye-patterns (1,7) RLL GMSK ............................. 85
E.3 Eye-patterns (2,7) RLL GMSK ............................. 86
E.4 Eye-patterns uncoded GMSK ............................... 87
E.5 Eye-patterns 2B1Q GMSK ................................ 88
E.6 PSDs (1,3) RLL GMSK ..................................... 89
E.7 PSDs (1,7) RLL GMSK ..................................... 90
E.8 PSDs (2,7) RLL GMSK ..................................... 91
E.9 PSDs uncoded GMSK ....................................... 92
E.10 PSDs 2B1Q GMSK ......................................... 93
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Encoded CPM system</td>
<td>11</td>
</tr>
<tr>
<td>2.2</td>
<td>FM transmission system</td>
<td>12</td>
</tr>
<tr>
<td>2.3</td>
<td>Configuration for characterization of FM channel</td>
<td>14</td>
</tr>
<tr>
<td>2.4</td>
<td>Basic canceller configuration</td>
<td>15</td>
</tr>
<tr>
<td>2.5</td>
<td>Transversal linear filter plus DC-tap</td>
<td>17</td>
</tr>
<tr>
<td>2.6</td>
<td>Simple recursive filter</td>
<td>17</td>
</tr>
<tr>
<td>2.7</td>
<td>Volterra-decomposed canceller structure</td>
<td>19</td>
</tr>
<tr>
<td>2.8</td>
<td>Generation of the channel symbols</td>
<td>30</td>
</tr>
<tr>
<td>2.9</td>
<td>State transition diagram of ((d,k)) code</td>
<td>31</td>
</tr>
<tr>
<td>2.10</td>
<td>Transition probabilities of ((d,k)) code</td>
<td>32</td>
</tr>
<tr>
<td>2.11</td>
<td>State transition diagram for the modulation sequence</td>
<td>32</td>
</tr>
<tr>
<td>2.12</td>
<td>State transition diagram for MFM</td>
<td>35</td>
</tr>
<tr>
<td>2.13</td>
<td>The 2B1Q encoder</td>
<td>35</td>
</tr>
<tr>
<td>C.1</td>
<td>FSM for MFM</td>
<td>62</td>
</tr>
<tr>
<td>C.2</td>
<td>Constraint graph for the IBM ((1,7)) code</td>
<td>63</td>
</tr>
<tr>
<td>C.3</td>
<td>Constraint graph for the IBM ((2,7)) code</td>
<td>64</td>
</tr>
<tr>
<td>D.1</td>
<td>State transition diagram</td>
<td>66</td>
</tr>
<tr>
<td>D.2</td>
<td>Area of integration of the autocorrelation function</td>
<td>72</td>
</tr>
</tbody>
</table>
## List of Tables

2.1 Illustration of the effect on $M'$ of restricting $Q$ ........................................... 20
2.2 Test configurations ................................................................................................. 25
2.3 Cancellers ............................................................................................................... 36
2.4 Estimated SNR values in dB ................................................................................. 36
3.1 Entropy of channel symbols and symbol rates of the applied codes .................. 39
4.1 Values for $B_g T_s$ ................................................................................................. 52
4.2 Multiplication factors .............................................................................................. 52
C.1 Data optimal (1,3) RLL code ................................................................................. 62
C.2 Data optimal (1,7) RLL code ................................................................................. 62
C.3 Data optimal (2,7) RLL code ................................................................................. 63
Chapter 1

Introduction

The representation of information in digital form is becoming more and more common nowadays. One can think of computers and digital audio equipment. And consequent, the transmission of digital information is subject of studies.

Transmission of (digital) information can be done in many ways. In this report wireless transmission using electromagnetic waves at radio frequencies is assumed. The radio frequency (RF) modulation technique to be used is frequency modulation (FM).

An RF channel is characterized by its allocated frequency band (which is determinated by a carrier frequency and a bandwidth) and the maximum transmission power. The allocation of RF channels is done under the auspices of the International Telecommunication Union (ITU). The ITU is a specialized agency of the United Nations (UN). Another characteristic is the nature of the noise processes which contribute to the power in the allocated band. Noise, in general, degrades the information bearing signal.

In practical systems the power spectral density (PSD) of the information bearing signal is not entirely limited to the allocated frequency band. As a result, a (small) fraction of the transmitted power ends up in the adjacent channels and interferes with the information bearing signals in those channels. To limit this so-called co-channel interference, the ITU demands that the PSD in the adjacent channels is below a given maximum. This leads to a more precise description of the RF channel: An RF channel is determined by a function of the frequency which denotes the maximum PSD at that frequency.

The constraints introduced to limit the co-channel interference are hard to meet in practical systems (K. Murota et al. [15]). The information bearing signal must have a relatively small spectrum and low spectral lobes. The combination of digital information, use of the FM technique and the demand for a narrow PSD has as a possible answer continuous phase modulation (CPM) techniques. See for example T. Aulin and C.E. Sundberg [5,6,20].

In a CPM scheme a baseband (rectangular) pulse train first is filtered. Then this filtered baseband signal is applied as the input of the FM modulator. Different filter types result in different CPM schemes. In this work CPM schemes containing a Gaussian filter
are studied. The resulting CPM scheme is known as Gaussian Minimum Shift Keying (GMSK). The original article on GMSK was written by K. Murota et al. [15]. A Gaussian filter is determined except for the parameter $B_g T_s$ which determines the $-3$dB bandwidth of the filter normalized at the symbol rate $T_s$. Murota [15] has shown that by varying $B_g T_s$ the bandwidth of the CPM signal can be tuned to a large range of values.

In most practical systems, digital information is not transmitted (or stored) as a sequence of independent data symbols. Instead, for reasons of a technical or economical nature, the sequence of symbols is first encoded. Popular coding schemes used with digital information sources are of the run-length-limited (RLL) type. For example, for the compact disk (CD), eight-to-fourteen modulation (EFM) is used, which is a member of the RLL coding class.

Encoding introduces correlation between the data symbols. Among others, P.K.M. Ho and P.J. McLane [10,11,12] have shown that the encoding of the data symbols generally influences the PSD of the CPM signal itself. This should not be a big surprise. As an illustration: It is known, for the most frequently used RLL schemes, that the resulting baseband signal (also known as the write signal) has the bulk of its energy at low frequencies (see e.g. A. Gallopoulos et al. [9]). It is also known that the FM modulator produces a more narrow spectrum if a baseband signal of low frequency is applied. Therefore it may be expected that an RLL coded baseband signal gives rise to a more narrow spectrum of the CPM signal than an uncoded baseband signal would do (N.B. it is assumed that the symbol rate on the channel in both schemes is the same.)

Another way to obtain a narrow spectrum of the CPM signal, is to decrease the symbol rate on the channel. Decreasing the symbol rate, normally, would imply that less information per unit time can be transmitted. The only way to compensate for this is to increase the information per symbol. The information per symbol can be increased by increasing the number of signal levels.

In this report Binary RLL coded GSMK, Uncoded Binary GMSK and 4-level GMSK are compared. The main questions treated in this report are: How can these schemes be compared? Which scheme is to be preferred?

An outline of the rest of this report is now given. In chapter 2 the encoded CPM transmission system is described. The chapter starts with an introduction. In section 2.2 the baseband channel of an FM transmission system is discussed. The discussion consists of two parts. The first part deals with the theoretical side. In the second part the characterization of practical channels is discussed. In section 2.4 the CPM transmission technique is discussed. Special attention is given to minimum shift keying (MSK) and Gaussian minimum shift keying (GMSK). The final section, 2.5, discusses coding schemes. This section is divided into a subsection about RLL coding and another about multi-level coding.

In chapter 3 an answer is formulated to the question: How to compare the different schemes? This discussion also tries to find a satisfactory answer to the question: What is the best scheme? Important in the comparison are the baseband behaviour and the power spectral density (PSD) of the CPM signal. The baseband behaviour the subject in section 3.2 the PSD is discussed in section 3.3.
In chapter 4 the results are presented in graphical form.

In chapter 5 conclusions are gathered based on the results.
Chapter 2

Description of the transmission system

2.1 Introduction

In this chapter the encoded CPM transmission system is discussed. As parts of the encoded CPM transmission system there is: the source, the encoder, the pulse train generator and the continuous phase modulation (CPM) modulator. A block diagram is shown in figure 2.1. The source generates a sequence of independent and equally distributed binary digits\(^1\) \{\(\alpha\}\}. The encoder transforms the sequence of independent binary digits \{\(\alpha\)\} in a possibly dependent sequence of channel symbols \{\(\gamma\)\}. The channel symbol sequence \{\(\gamma\)\} is then applied to the "pulse train generator". The "pulse train generator" uses the channel symbol sequence \{\(\gamma\)\} to generate the continuous signal \(W(t)\), which is applied to the CPM modulator. In figure 2.1, the CPM modulator it is shown that the CPM modulator consists of a premodulation filter and an FM modulator.

In section 2.2 the baseband channel of a frequency modulation (FM) transmission system is discussed. The discussion is divided into two parts. First some theoretical properties are stated, such as the channel transfer function and the shape of the baseband noise power spectral density (PSD). In the second part practical FM transmission systems are examined. The baseband channels of these systems were characterized using techniques based on echo-canceling. These techniques are described in an article by J.W.M. Bergmans et al. \[7\].

In section 2.4 the theoretical background of continuous phase modulation (CPM) is given. Special attention is given to two types of CPM schemes. The first is minimum shift keying (MSK), which is the most simple CPM scheme. The second scheme is Gaussian minimum shift keying (GMSK). The MSK scheme is used as a reference. All schemes investigated for this work are based on the GMSK scheme.

\(^1\)A sequence of random (one-dimensional) variables, \(\ldots x_{-1} x_0 x_1 \ldots\), is denoted by \{\(x\)\}. 
In section 2.5 premodulation coding is discussed. This is the type of coding performed on the data before the data is fed to the CPM system. The types of coding discussed are: run-length-limited (RLL) coding and a binary-to-four level coding technique.

2.2 The baseband channel

2.2.1 Theoretical discussion of the baseband channel of an FM transmission system

In this section some of the properties of the baseband channel of an ideal FM transmission system are stated. These properties can be found in any textbook that covers the subject (see e.g. K.S. Shanmugam [19]).

A block diagram of an idealized FM transmission system is shown in figure 2.2.

The FM modulator is assumed to be ideal. Its output is

\[ S(t) = A \cos\{2\pi f_d t + \phi(t)\}, \quad (2.1) \]

where \( A \) is a constant amplitude, \( f_c \) is the carrier frequency and \( \phi(t) \) is the phase deviation of the carrier. The phase deviation is a function of the message signal \( x(t) \):

\[ \phi(t) = k_f \int_{-\infty}^{t} x(\tau) d\tau, \quad (2.2) \]

in which \( k_f \) is the frequency deviation constant and furthermore it is assumed that

\[ \phi(t) \leq \pi. \]

Figure 2.1: Encoded CPM system
The RF channel is assumed to be distortionless, which means that the output of the channel is an attenuated and possibly delayed version of the input signal. Furthermore it is assumed that the noise process represented by $n_i(t)$ is additive and that it is white in the passband. The bandpass filter (BPF) in figure 2.2 eliminates the out-of-band noise and passes the FM signal undistorted.

The FM demodulator or discriminator is also taken to be ideal. Its input is

$$y(t) = a \cos(2\pi f_r t + \phi(t)) + n(t),$$  \hspace{1cm} (2.3)$$

where $a$ is such that $a/A$ represents the attenuation by the HF channel. The noise term $n(t)$ is the band limited version of $n_i(t)$. The process $n(t)$ represents zero mean Gaussian noise with a power spectral density (PSD) function

$$P_n(f) = \begin{cases} \eta/2, & |f - f_c| < B_T/2 \\ 0, & \text{elsewhere.} \end{cases}$$  \hspace{1cm} (2.4)$$

Here $\eta$ is the PSD of the noise and $B_T$ represents the transmission bandwidth of the FM signal (in Hertz).

It can be shown (see e.g. Shanmugam [19]) that the phase angle of the signal at the input of the demodulator (eq. 2.3) is equal to

$$\theta_y(t) \approx \phi(t) + \frac{r_n(t)}{a} \sin[\theta_n(t) - \phi(t)],$$  \hspace{1cm} (2.5)$$

where $r_n(t)$ is the current modulus and $\theta_n(t)$ represents the current phase angle of the band limited noise signal $n(t)$. Equation 2.5 is only valid when $a^2 \gg E[n^2(t)]$, where $E[.]$ denotes expected value.

The discriminator extracts an approximation of the input signal $x(t)$ by differentiation of the phase angle $\theta_y(t)$:

![Figure 2.2: FM transmission system.](image-url)
\[ \tilde{x}(t) = k_d \frac{df}{dt} [\theta_y(t)] \]
\[ = x(t) + k_d \frac{df}{dt} [r_n(t) \sin[\theta_n(t)]], \quad (2.6) \]

where \( k_d \) is the discriminator sensitivity. The last equation is obtained by choosing \( k_f k_d = 1 \) and setting \( \phi(t) = 0 \) (which is allowed, see Shanmugam [19]). The low-pass filter (LPF) removes all components outside the message band but passes the message signal \( x(t) \) undistorted. Thus, the channel transfer function or the idealized FM transmission system is

\[ H(f) = \begin{cases} 
1, & |f| \leq B_b \\
0, & \text{elsewhere}, \end{cases} \quad (2.7) \]

where \( B_b \) is the baseband bandwidth.

Before the expression for the PSD of the noise in the baseband channel is posed, three remarks are made. The term \( r_n(t) \sin(\theta_n(t)) \), which is the amplitude of the sine component of the so-called quadrature representation of the noise \( n(t) \), has a PSD \( \eta \) for \( |f| < B_T/2 \). Differentiation is represented in the frequency domain by a multiplication by \( j2\pi f \). And finally the LPF rejects all signal components that lie outside the message band. The output noise PSD is given by

\[ P_{n_o}(f) = \begin{cases} 
\frac{1}{\pi \eta^2} (2\pi f)^2 \eta, & |f| < B_b/2 \\
0, & \text{elsewhere}, \end{cases} \quad (2.8) \]

where \( n_o(t) \) represents the noise process at the output. Equation 2.8 shows that the noise spectrum is parabolic.

### 2.3 Canceller-based channel characterization

#### 2.3.1 Introduction

In the previous section it was shown that in an FM transmission system the message signal \( x(t) \) is passed undistorted. Furthermore it was shown that the shape of the noise PSD in the baseband of the transmission system is parabolic. To get to these conclusions several assumptions were made in order to simplify the mathematics. These assumptions idealize the FM transmission system, possibly alienating it from a practical system.

Therefore, as a first assignment, the baseband behaviour of a practical FM transmission system was studied. The results are gathered in this section. The most important questions where: Does the channel introduce linear intersymbol interference (ISI)? Does the channel introduce nonlinear ISI? What does the (baseband) noise PSD look like? The procedure of determining these characteristics is called channel characterization.
The idea behind the characterization method is that it must be possible to estimate the relevant channel characteristics of a practical system by examining its input and the accompanying output. To generate a series of channel inputs and corresponding channel outputs, the test configuration of figure 2.3 was used.

The channel characteristics might depend on the actual signaling scheme used. However, at the time of this characterization it was not known how the final implementation of the FM data transmission system was going to be. So it was not possible at that time to test the channel in the final mode of operation. Some tests had already been performed using an eight-level pulse amplitude modulation system (PAM system). These results were used to perform the characterization.

Figure 2.3 shows the PAM system. The computer generates an eight-level data signal with a symbol rate of 128 kbaud. The data signal is converted to the analog domain by the DAC. Before transmission the signal is filtered to strip it from frequency components above the message band. This is done by a transmit low-pass filter (LPF\(_t\)). Then the signal is transmitted by the FM transmitter. The received signal is demodulated by the FM receiver. A receive low-pass filter (LPF\(_r\)) removes the out of message band signals. Finally the analog signal is sampled and the received samples are stored by the computer.

To characterize the channel, a method described by Bergmans et al. [7] was used. Bergmans et al. used echo cancellation techniques to estimate the channel impulse response and the noise PSD. Figure 2.4 shows the basic canceller configuration. The channel output \( r_n \) is a function of the previous and current channel inputs and the channel noise \( u_n \). It is assumed that the channel noise is uncorrelated with the elements of the channel input sequence\(^2\) \( \{a\} \). The channel input symbols are applied to the channel one after another. When channel symbols up to and including \( a_n \) are applied the channel produces the output \( r_n \). The channel outputs are of the form

\[^2\text{A sequence of random (one-dimensional) variables, \( \ldots x_{-1} x_0 x_1 \ldots \), is denoted by \( \{x\} \).}\]

![Figure 2.3: Configuration for characterization of FM channel.](image)
where \( s_n \) is a deterministic function of the channel input sequence \( \{a\} \). Since the channel is assumed to be causal, only the channel input symbols up to and including \( a_n \) can contribute to \( r_n \). It is also assumed that the channel is time independent. As a result the zero noise output \( s_n \) of the channel may be represented as

\[
    s_n = f(a_n),
\]

(2.10)

where \( a_n \) represents the sequence of input symbols up to and including \( a_n \).

How does canceller-based channel characterization work? The channel input symbols sequence \( \{a\} \) is applied to an adaptive filter. When the symbols up to \( a_n \) are applied, this filter, the canceller, produces an output \( s'_n = f'(a_n) \). The canceller output is subtracted from the channel output. The resulting error signal has the form

\[
    \epsilon_n = f(a_n) - f'(a_n) + u_n.
\]

(2.11)

Since data and noise are uncorrelated, the power \( U' \) of \( \epsilon_n \) is minimized by choosing \( f'(\cdot) \) to equal \( f(\cdot) \). This suggests the use of an adaptation algorithm which monitors \( U' \) by observing \( \epsilon_n \), and thus adjusts \( f'(\cdot) \) until the minimum of \( U' \) is reached. Upon completion of this procedure, \( f'(\cdot) \) specifies \( f(\cdot) \), and \( \epsilon_n \) describes \( u_n \). It has been shown by Bergmans et al. [7] that, in the absence of nonstationarities, arbitrarily precise reconstruction of the channel transfer function can be guaranteed, provided that enough data samples are available.

---

3 A sequence of random (one-dimensional) variables, \( \ldots x_{n-2} x_{n-1} x_n \), is denoted by \( x_n \).
The adaption algorithm used in the article is LMS. The advantages of this algorithm are: it is simple, its behaviour has been thoroughly analyzed, and its implementation does not depend on the precise data correlation structure. Some of the important properties of the LMS algorithm is discussed later in this report.

2.3.2 Canceller structures

In this section three different structures for the canceller in figure 2.4 are discussed. The most simple structure is obtained if the channel behaviour is approximated by a linear function of a finite number of channel inputs. For this situation a transversal linear filter structure can be used to implement the canceller. For the second structure the transversal filter is extended with a recursive section. The purpose of this extension is to provide the canceller impulse response with an exponentially decreasing tail. Finally a nonlinear structure based on Volterra series expansion is used. Each canceller contains a number of adaptive coefficients called taps. To be able to cancel a DC-offset all cancellers were extended with a DC-tap. Of course, then the canceller is not linear anymore. In this report, however, the linear canceller extended with a DC-tap is still referred to as the linear canceller.

If each channel output is approximated by a DC-level increased with a linear function of a finite number of channel inputs then the output signal $r_n$ can be written as

$$ r_n = f_{DC} + a_n^T f + u_n. \tag{2.12} $$

Where $f_{DC}$ represents the DC-level and

$$ a_n = [a_n, \ldots, a_{n-M+1}]^T \tag{2.13} $$

is the vector of the current and $M-1$ most recent data symbols. Furthermore $[\cdot]^T$ denotes vector transposition,

$$ f = [f_0, \ldots, f_{M-1}]^T \tag{2.14} $$

is the channel impulse vector. The length of each vector is $M$, which represents the memory length of the channel. The time sequence $f_0, \ldots, f_{M-1}$ is the impulse response of the channel; its Fourier transform is the channel transfer function. The convolution in equation 2.12 can be reproduced by means of an adaptive linear filter extended with a DC-tap, whose output signal $s'_n$ can be written as

$$ s'_n = f'_{DC} + a_n^T f'_n. \tag{2.15} $$

In figure 2.5 a transversal linear filter is shown which is a possible implementation of the linear canceller.
It is obvious that the memory length $M$ of the canceller should be at least equal to the memory length of the channel. The channel length might be quite large, it might even be infinite. To cover for such situations $M$ has to be large. Making $M$ large, however, has the drawback that the characterization with the LMS algorithm becomes less accurate. In general, the more taps the canceller has, the harder it is for the LMS algorithm to adapt the filter fast and accurate. In a practical system one often encounters an impulse response which after a number of time units $M_t$, shows an exponential decreasing tail according to

$$f_{i+1} = \beta f_i, \ |\beta| < 1, \ i \geq M_t. \tag{2.16}$$

By extending the transversal linear filter of figure 2.5 with a recursive section the second implementation of the canceller is obtained. In figure 2.6 this modification is shown. Note that only one tap ($\beta$) is needed to generate the exponential tail.

From figure 2.6 one can derive that

$$s'_n = f'_{DC} + \sum_{i=0}^{M-2} f'_i a_{n-i} + z_n \tag{2.17}$$
and

\[ z_n = f'_{M-1}a_{n-M+1} + \beta z_{n-1}. \quad (2.18) \]

Use of equation 2.18 in equation 2.17 gives

\[ s'_n = f'_{DC} + \sum_{i=0}^{M-1} f'_i a_{n-i} + f'_{M-1} \sum_{i=M}^{\infty} \beta^{M+1-i} a_{n-i}. \quad (2.19) \]

Finally, to estimate the nonlinearity of the channel, a nonlinear canceller structure was used. In Bergmans et al. [7] it is proposed to use a Volterra series expansion. For example for a channel with a memory length \( M = 2 \) and input alphabet \{-1,1\}, the canceller response can be written as

\[ f'(a_n) = f'_{DC} + f'_0 a_n + f'_1 a_{n-1} + f'_2 a_{n-2} + f'_{01} a_n a_{n-1} + f'_{02} a_n a_{n-2} + f'_{12} a_{n-1} a_{n-2} + f'_{012} a_n a_{n-1} a_{n-2}. \quad (2.20) \]

One may wonder why a term like \( f'_{00} a_n^2 \) is not present in equation 2.20. The answer is simple: for \( a_n = -1 \) and for \( a_n = +1, a_n^2 = +1 \) making the term \( f'_{00} a_n^2 \) equal to \( f'_{00} \). This makes \( f'_{DC} \) and \( f'_{00} \) dependent. More generally: The input vector \( a_n \) can take one of \( 2^3 = 8 \) different values. This means that equation 2.20 is not allowed to contain more than 8 parameters \( f' \). Thus a canceller for this channel should not contain more than 8 taps.

In the previous section it was already mentioned that input and corresponding output samples of an eighth-level PAM-system were used. If the memory length is assumed to be \( M \) then the number of input vectors \( a_n \) is equal to \( 8^M \). If terms like \( f'_{00} a_n^2, f'_{000} a_n^3, f'_{001} a_n^2 a_{n-1} \) etc. are allowed to be a part of the function describing the canceller, then, to prevent overdimensioning of the canceller, at least the following inequality must be satisfied

\[ \left( \begin{array}{c} M + P \\ P \end{array} \right) \leq 8^M, \quad (2.21) \]

where \( P \) is the maximum order of nonlinearity. The left-hand-side of inequality 2.21 represents the number of taps of a canceller with a memory length of \( M \) and a maximum order of nonlinearity \( P \).

In practice only some of the nonlinear coefficients may be significant. The remainder can then be neglected to improve tracking and numerical efficiency. The procedure to determine the degree and type of channel nonlinearities, sugested by Bergmans et al. [7] is to start with only the DC-tap and the linear taps, and gradually increase the maximum
order of nonlinearities $P$ until the power $U'$ of $c_n$ no longer decreases. However for large $M$ this process rapidly leads to an unpractical total number of coefficients $M'$. In the article it is suggested to limit the number of coefficients by restricting the temporal span $Q$ of the nonlinearities. The concept temporal span of nonlinearities can be explained best using an example. Suppose one of the terms in the canceller response (like in equation 2.20) is

$$f_{157}a_{n-1}a_{n-5}a_{n-7}.$$ 

The oldest input symbol that contributes to this term is $a_{n-7}$, the most recent one is $a_{n-1}$. The temporal span of nonlinearities denotes the time span between those two input symbols. In the example, it is the number of input symbols in the sequence $[a_{n-7}, a_{n-6}, \ldots, a_{n-1}]$, which is 7. Table 2.1 exemplifies the resulting reduction of $M'$ for canceller structures like the one in figure 2.7.

To account for canceller nonlinearities, the vectors $f'$ and $f$ can be extended to include the nonlinear canceller coefficients and corresponding channel coefficients, respectively.
<table>
<thead>
<tr>
<th>Maximum order of nonlinearity</th>
<th>Total number of coefficients ( M' ) (including DC tap)</th>
</tr>
</thead>
<tbody>
<tr>
<td>( P )</td>
<td>( M = 5 )</td>
</tr>
<tr>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>21</td>
</tr>
<tr>
<td>3</td>
<td>56</td>
</tr>
<tr>
<td>4</td>
<td>126</td>
</tr>
<tr>
<td>5</td>
<td>252</td>
</tr>
</tbody>
</table>

Table 2.1: Illustration of the effect on \( M' \) of restricting \( Q \).

2.3.3 LMS-based channel characterization

In this section some of the characterization capabilities of the LMS-based canceller are stated. One result is that accurate noise estimation is much more difficult than accurate channel identification. Furthermore an expression for the excess mean-square error due to imperfect settings of the canceller coefficients is given. This expression is valid after the adaption of the canceller. It is assumed that \( a_n, f_n \) and \( u_n \) are statistically independent stationary processes. Initially, all variations of the channel parameters are neglected. We found that this is not allowable (see the section about 'DC wander'). In the sequel it is assumed that the data is normalized to have a power \( E[a_n^2] = 1 \).

Some definitions

As in Bergmans et al. [7], the signal to-noise-ratio of the channel is defined as

\[
SNR = \frac{S}{U},
\]

where

\[
S = E[s_n^2]
\]

(2.23)

is the power of the received data component \( s_n \), while

\[
U = E[u_n^2]
\]

(2.24)

is the power of the noise signal \( u_n \) (see fig. 2.4). If the channel output \( s_n \) can be approximated by \( s_n = a_n^T f_n \), equation 2.23 can be written as

\[
S = E[f_n^T R f_n],
\]

(2.25)
where

$$R = E[a_n a_n^T]$$

(2.26)

is the data autocorrelation matrix.

The error signal can be written as

$$e_n = r_n - s'_n = s_n - s'_n + u_n = a_n^T(f_n - f'_n) + u_n.$$

(2.27)

Upon completion of the adaption of the canceller, $e_n$ is used as an estimate of $u_n$. The estimated signal-to-noise ratio $SNR'$ is given by

$$SNR' = S'/U',$$

(2.28)

where $S'$ is the power of the canceller output signal $s'_n$ and $U'$ is the power of $e_n$. Using equation 2.27 and the fact that $a_n$, $f_n$ and $u_n$ are independent gives

$$U' = E[e_n^2] = e + U$$

(2.29)

where

$$e \triangleq E[(f'_n - f_n)^T R (f'_n - f_n)]$$

(2.30)

is the excess mean-square error due to imperfect settings of the canceller coefficients. It is obvious that accurate noise characterization requires that $e \ll U$.

To be able to compare $f_n$ and $f'_n$, the mean-square misadjustment is introduced, which is defined as

$$\Delta \triangleq E[(f'_n - f_n)^2] = E[(f'_n - f_n)^T (f'_n - f_n)].$$

(2.31)

Ideally, $\Delta$ should be much smaller than the $L_2$ norm

$$F = E[|f_n|^2] = E[f_n^T f_n]$$

(2.32)

of $f_n$.

If the data is uncorrelated then $R$ is the $M'$ by $M'$ identity matrix. In this case $S = F$ and $e = \Delta$. Use of equation 2.22, and assuming that the data is uncorrelated, it is not difficult to show that the normalized inaccuracies $e/U$ and $\Delta/F$ satisfy
\[ \frac{\xi}{U} = \frac{\Delta}{F'} SNR. \]  

(2.33)

The last equation shows that for realistic SNRs accurate noise characterization is much more difficult than accurate channel identification. If the data is uncorrelated then equation 2.33 is no longer valid. But, except for untypical situations accurate noise characterization is the more difficult task.

**LMS properties**

The LMS algorithm updates \( f_i \) recursively according to

\[ f_{n+1} = f_n + \mu e_n a_n, \]

(2.34)

where \( \mu \) is an adaptation constant. The adaptation process generally consists of two phases:

1. a transition phase, in which an initial misadjustment of the coefficients is gradually lowered, and

2. a steady-state phase, in which the transient errors have become negligible compared with the tracking errors and random fluctuations of the coefficients about their average settings.

In the steady-state phase, the canceller has become a model of the channel, and the error signal \( e_n \) resembles the (baseband) channel noise \( u_n \).

The steady state error \( \varepsilon \) (equation 2.29 consists of two independent contributions:

\[ \varepsilon = \varepsilon_g + \varepsilon_l, \]  

(2.35)

where \( \varepsilon_g \) is the gradient error due to random fluctuations of the coefficients around their average settings and \( \varepsilon_l \) is the lag error due to imperfect tracking. The gradient error can be approximated by (see Bergmans et al. [7])

\[ \varepsilon_g \approx \frac{M'\mu}{2 - M'\mu} U. \]  

(2.36)

Note that more taps (the number of taps is \( M' \)) increase the gradient error. Also, increase of the convergence speed of the adaption process by increasing \( \mu \), increases the the gradient error. The lag error is exclusively due to channel parameter variations. If the channel parameters vary in time, the canceller tries to follow these changes. In general, as a result, the canceller transfer function differs from the channel transfer function, which gives rise to
the lag error. If $\mu$ is very small, then (at a certain point) the channel parameter variations cannot be tracked, and their full power is reflected in $\varepsilon_i$.

Summarizing, the number of taps should be kept small. The adaptation constant should not be too large, in order to prevent a large gradient error, and it should not be too small, in order to prevent a large lag error. In a practical situation one can vary $\mu$ until the signal $e_n$ has minimum energy.

2.3.4 Results

In this section the results obtained by performing a series of characterizations are discussed. Since, in the steady-state phase of the adaptation process, the error signal $e_n$ resembles the noise signal $u_n$, the PSD of $u_n$ can be estimated by calculating the PSD of $e_n$. This was done by first calculating the autocorrelation function, and subsequently determine its Fourier transform. The autocorrelation function is calculated using the formula

$$R_e(k) = E[e_n \cdot e_{n+k}], \quad (2.37)$$

where $E[.]$ denotes the expectation over all $n$. The PSD is then found using

$$E(e^{j\omega}) = \sum_{k=-\infty}^{\infty} R_e(k)e^{-j\omega k}. \quad (2.38)$$

In the appendix one can find plots of $|E(e^{j\omega})|$. The channel transfer function is estimated as the transfer function of the canceller. The last one is calculated using the formula

$$H(e^{j\omega}) = \sum_{k=0}^{\infty} f_k e^{-j\omega k}. \quad (2.39)$$

In the appendix, one can find plots of $|H(e^{j\omega})|$. 

To be able to compare the baseband behaviour of different FM channels, the input and output sequences were normalized. First the DC component was removed and secondly the power was normalized to 1. The DC component is

$$d = \frac{\sum_{n=0}^{n} r_n}{\# r_n}, \quad (2.40)$$

where $\#$ denotes 'number of'. Then, a scale factor

$$s = \sqrt{\frac{\sum_{n}(r_n - d)^2}{\# r_n}}, \quad (2.41)$$
can be calculated. The normalized channel output is

\[ r_n^* = \frac{(r_n - d)}{s}. \]  

(2.42)

The observant reader may wonder why the the presented canceller structures contain a DC-tap, since a possibly present DC-component is removed before characterization. This is explained shortly.

The transfer function, as defined in equation 2.39, is only valid for a linear canceller. In case the baseband channel of the FM system is (slightly) nonlinear, then the transfer function based on the values of the linear taps can still be used as a first order estimate of the channel transfer function.

DC-wander

The first characterizations were done using the transversal canceller of figure 2.5. The estimated noise PSD showed an unexpected large value for DC. Decreasing \( f_L \) from \( 2^{-9} \) to \( 2^{-4} \) made this large peak at DC disappear completely, while on the other hand the noise PSD for other frequencies, increased as expected (see equations 2.29, 2.30 and 2.36). The cause of the large DC-value in the noise PSD for small values of \( f_L \) is what is called 'DC-wander'.

Remember that the input and output sequences were normalized before channel characterization was done. Thus, when the whole normalized output sequence is regarded, no DC-component is present. But here, as a result of channel parameter variations, the output sequence contained a slowly varying component, uncorrelated with the input sequence. The frequency of the variations is much smaller than the symbol rate on the channel. Since this component is uncorrelated to the input sequence, it must be cancelled by the DC-tap. This situation is denoted by ‘DC-wander’. If \( f_L \) is too small to follow the variations then their full power is reflected in the noise PSD (see Bergmans et. al. [7]). Which explains the observed.

To be able to perform accurate channel characterization a separate adaption constant, \( \mu_{DC} \), was used for the DC-tap. The value for \( \mu_{DC} \) was chosen such that the DC-tap could be updated fast enough to follow the slow changes in the DC-level due to the DC-wander. Since \( f_L \) wasn’t used to adapt the DC-tap, it could be made much smaller in order to minimize the gradient error.

The characterizations

For the characterizations, there were six different channel output files with the corresponding input files. The first four files were generated with a transmission system of good quality. The last two files were generated using a more economical transmission system. These two transmission systems will be referred to as ‘system 1’ and ‘system 2’.
respectively. Files 3, 4 and 5 were generated by applying pre-emphasis. The pre-emphasis filter was implemented as a digital filter in the computer. This filter has the system function

\[ H_p(z) = 1 - \frac{z^{-1}}{2}. \]  

(2.43)

The amplitude response of the filter is

\[ |H_p(e^{j\theta})| = \left| \frac{e^{j\theta} - 0.5}{e^{j\theta}} \right| = \sqrt{\frac{5}{4} - \cos \theta}. \]  

(2.44)

This equation shows that \( H_p[z] \) amplifies high frequency and suppresses low frequency components of the message signal. The first four files differ in the signal strength at the HF input of the receiver. File 1 and 3 have a low signal strength of 40 \( \mu \)V and where files 2 and 4 have a high signal strength of 1 \( m \)V. Table 2.2 shows the different configurations used to generate the test files. The entry \textit{system} determines the transmission system used, where 1 and 2 refer to the professional and the economical system, respectively. If the entry \textit{pre-emp.} denotes pre-emphasis. The entry \textit{strength} denotes the signal strength. The economical transmission system was only tested with a signal strength of 40 \( m \)V.

<table>
<thead>
<tr>
<th></th>
<th>file 1</th>
<th>file 2</th>
<th>file 3</th>
<th>file 4</th>
<th>file 5</th>
<th>file 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>system</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>pre-emph.</td>
<td>no</td>
<td>no</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>no</td>
</tr>
<tr>
<td>strength</td>
<td>40 ( \mu )V</td>
<td>1 ( m )V</td>
<td>40 ( \mu )V</td>
<td>1 ( m )V</td>
<td>40 ( \mu )V</td>
<td>40 ( \mu )V</td>
</tr>
</tbody>
</table>

Table 2.2: Test configurations.

Table 2.3 describes the different cancellers used. Cancellers 1 to 3 are linear cancellers. To implement cancellers 1 and 2 the Finite Impulse Response (FIR) filter of figure 2.5 was used. Canceller 3 was implemented using the recursive filter of figure 2.6. Cancellers 4 to 9 are nonlinear. These cancellers were implemented using the Voltera decomposed canceller structure of figure 2.7.

The better the canceller immitates the channel, the smaller the excess mean-square error due to the imperfect immitation is. As a result the estimated SNR, based on the error signal \( e_n \), drops. If the canceller perfectly immitates the channel then the estimated SNR is equal to the SNR on the channel. In table 2.4 the estimated SNR values found with the characterization using the different files and different cancellers are gathered.

Comparing the result of file 1 with file 3, file 2 with file 4 and file 6 with file 5 (the second file of each pair is the file with preemphasis), shows that preemphasis does not give a significant improvement. Preemphasis seems to make characterization of the nonlinearities
in the channel more difficult. The SNRs for files 4 and 5 show a 6.2 dB and 2.6 dB improvement, respectively, for the preemphasis files, by use of the nonlinear canceller 1 over the linear canceller 7. The corresponding files without preemphasis (2 and 6) show an improvement of 9.5 dB and 6.8 dB, respectively. This concludes the discussion of the results obtained with files 3, 4 and 5.

After the FIR filter is adapted using file 1, it has the impulse response as shown in appendix A, page 57. This impulse response shows that the channel suffers from severe linear ISI. This is an important difference with the results obtained by the theoretical discussion; in theory the message signal is transmitted undistorted. The noise spectrum has the expected parabolic shape except for a low noise floor (see appendix A, page 58. This noise floor could be the result of system noise added before modulation in the transmitter or after demodulation in the receiver. If the transmission system 1 is used with low power (file 1), then from table 2.4, a nonlinear canceller does not improve the estimated SNR. The conclusion is that in this mode of operation the baseband channel of transmission system 1 is essentially linear. Page 59, appendix A shows the amplitude response of the adapted FIR filter.

The SNR in the baseband of an FM system is proportional to the reciprocal value of the power of the transmitted signal at the input of the receiver (see Shanmugam [19]). Increasing the signal strength from 40 μV to 1 mV, which is done to generate file 2, should increase the SNR by 28 dB. Table 2.4, however, shows an increase of only 5.4 dB. According to the data sheets of the receiver of system 1, an input strength of 1 mV means that the receiver is in saturation. The sheets predict an improvement of 5 dB, which is close to the results of table 2.4.

Table 2.4 shows that for the receiver of system 1 in saturation (file 2), the estimated SNR is greatly improved if a nonlinear canceller structure is used. An improvement of almost 10 dB is possible, which corresponds theoretically to a 2 to 4 times higher data rate. We found that if linear taps $f_x$ and $f_y$ are large then the tap $f_{xy}$ of order 2 is also likely to be important.

Comparing the data sheets of the receiver of transmission system 1 with those of the receiver of transmission system 2 shows that the only difference between the two receivers is that the first one shows less harmonic distortion (a factor 3). The baseband SNR as function of the input signal strength is the same for both receivers. Comparison of the entries in table 2.4 shows that even for small signal strength, transmission system 2, introduces nonlinear distortion. Table 2.4 reveals further that, at least for the economical transmission system (file 6), deleting small taps does not decrease the estimated SNR. Increasing the memory length beyond 16 does not give a better characterization. Extending a FIR filter of length 16 with an exponential tale is no improvement.
2.4 Continuous Phase Modulation

Continuous Phase Modulation (CPM) denotes a class of digital modulation schemes of the form $A \cos\{2\pi f_c t + \phi(t)\}$ where $A$ is a constant amplitude and $f_c$ is the carrier frequency. The phase deviation $\phi(t)$ is continuous and is a function of the input data. Due to the continuous nature of $\phi(t)$, the PSD of CPM schemes are relatively narrow with low spectral side lobes. This property is desirable for achieving low interference levels in adjacent channels. Furthermore, good error performance can be obtained if a coherent Maximum Likelihood Sequence Estimator (MLSE) is used as detector.

The most simple CPM scheme is Minimum Shift Keying (MSK). Other schemes, like Gaussian Minimum Shift Keying (GMSK), are better in terms of power-bandwidth performance, at the cost of increased signal processing. A survey article on the most popular CPM schemes has been written by C.-E.W. Sundberg [20]. In this article, although not explicitly stated, it is assumed that the input data symbols of the CPM schemes are equiprobable and uncorrelated! (The case of correlated input symbols is discussed later in this work.)

Coding of the input data stream introduces correlation. This correlation can change the characteristics of a CPM signal completely. See e.g. the articles by P.K.M. Ho and P.J. McLane [10,11,12].

Figure 2.1 shows the block diagram that can be used to generate all CPM schemes mentioned in this report.

The source in figure 2.1 generates an infinitely long sequence of binary (0,1) symbols, denoted by $\{a\}$, at a rate $1/T_b$ bits/s, where $T_b$ is the bit duration time. It is assumed that all symbols $a_i$ are independent and identically distributed. The encoder projects the input stream $\{a\}$ on the channel input sequence $\{b\}$. The channel symbols are elements of the set $\{\pm1, \pm3, \ldots, \pm(Q-1)\}$. The rate at which channel symbols are produced is $1/T_s$, where $T_s$ is the symbol duration time. Note that, due to the encoding process, $T_s$ may be different from $T_b$.

The CPM signal associated with the sequence $\{\gamma\}$ is

$$S(t, \{\gamma\}) = \sqrt{\frac{2E}{T_s}} \cos\{2\pi f_c t + \phi(t, \{\gamma\}) + \phi_0\}, \quad (2.45)$$

where the transmitted information is contained in the phase deviation

$$\phi(t, \{\gamma\}) = \pi h \sum_{n=-\infty}^{\infty} \gamma_n g(t - nT_s), \quad (2.46)$$

with phase response function
\[ g(t) = \int_{-\infty}^{t} f(\tau)d\tau. \] (2.47)

In equation 2.45-2.47, \( E \) is the symbol energy, \( h \) is the modulation index and \( \phi_0 \) is a constant phase shift. The function \( f(t) \) is the so-called frequency pulse function, which is the convolution of a rectangular pulse

\[ p(t) = \begin{cases} 
\frac{1}{T_s}, & -T_s/2 \leq t \leq T_s/2 \\
0, & \text{otherwise}, 
\end{cases} \] (2.48)

with the impulse response of the premodulation filter (see fig. 2.1).

As in Sundberg [20] it is assumed that \( f(t) \) is strictly limited to the interval \( [0, LT_s] \), where \( L \) is the length of the interval in units \( T_s \). Furthermore, in this report, \( f(t) \) is normalized such that

\[ g(LT_s) = \int_{0}^{LT_s} f(t)dt = 1. \] (2.49)

As a result of this normalization, the net phase change caused by the symbol \( \gamma_s \) is \( \pi h \gamma_s \), which agrees with the definition of \( h \) as the modulation index. Different choices of \( f(t) \), \( h \) and/or \( Q \) result in different CPM schemes.

The frequency pulse functions for a rectangular pulse of duration \( L \) symbol intervals \( T_s \) is defined as

\[ f(t) = \begin{cases} 
\frac{1}{LT_s}, & 0 \leq t \leq LT_s \\
0, & \text{otherwise}. 
\end{cases} \] (2.50)

The CPM scheme using this frequency pulse function of length \( L \) is denoted by the abbreviation LREC. If \( L = 1 \) then 1REC CPM is obtained. The 1REC CPM scheme is also known in literature as continuous phase frequency shift keying (CPFSK). If apart from \( L = 1 \), one chooses \( h = 0.5 \) and \( Q = 2 \), then the most simple CPM scheme is obtained, which is called: minimum shift keying (MSK). MSK is also known as fast frequency shift keying (FFSK). This scheme has been studied thoroughly. One of the results of these studies is a closed form expression for the power spectral density (PSD):

\[ P_s(f) = \frac{E}{2T_s} \{ P_v(f - f_c) + P_v(-f - f_c) \}, \] (2.51)

where

---

4This small deviation from the definitions of Sundberg was made to be consistent with the definitions of Ho in his articles discussed in chapter 3.
This result was used to judge the PSD calculation programs used to generate the results of appendix E. Note that \( P_v(f) \) integrates to unity.

Another frequency pulse function is obtained if the premodulation filter is a Gaussian filter. The resulting CPM scheme is denoted by Gaussian minimum shift keying (GMSK). The magnitude of the frequency response of the Gaussian filter is defined by

\[
|H(f)| = \exp\left[-\frac{\ln2}{2} \frac{f}{B_g}\right], \tag{2.53}
\]

where \( B_g \) is the -3dB bandwidth.

The Gaussian filter has a linear phase response which corresponds to a constant group delay. The actual group delay is not significant, therefore it is assumed to be zero. Now the impulse response \( h(t) \) of the Gaussian filter can be calculated by applying the inverse Fourier transform. The impulse response is

\[
h(t) = \sqrt{\frac{2\pi}{\ln2}} B_g \exp\left(-\frac{2\pi^2 B_g^2 t}{\ln2}\right). \tag{2.54}
\]

Convolution of the pulse \( p(t) \) (eq. 2.48) with \( h(t) \) gives the frequency pulse function for the GMSK scheme:

\[
f(t) = \frac{1}{T_s} \left[ Q\left(2\pi B_g \frac{t-T_s}{\sqrt{\ln2}}\right) - Q\left(2\pi B_g \frac{t+T_s}{\sqrt{\ln2}}\right) \right] \quad 0 \leq B_g T_s \leq \infty, \tag{2.55}
\]

where \( Q(t) \) is defined as

\[
Q(t) = \int_t^\infty \frac{1}{\sqrt{2\pi}} \exp(-y^2/2)dy. \tag{2.56}
\]

The GMSK scheme, first described by K.Murota et al. [15], is actually a set of schemes. Different choices for the parameter \( B_g T_s \) result in different frequency pulse functions (see appendix B).

Note that for \( B_g T_s = \infty \) the GMSK scheme simplifies to the MSK scheme. Murota showed that decreasing \( B_g T_s \) narrows the spectrum of the resulting CPM signal, at the cost of the increased of intersymbol interference (ISI).
2.5 Premodulation coding

2.5.1 Introduction

The main objective of this report is to compare three GMSK systems. The first and most simple is binary uncoded (NRZ) GMSK, the second is GMSK fed with the output of a run-length-limited (RLL) encoder, the third is GMSK fed with the output of a 2-binary-to-1-quaternary (2B1Q) encoder.

In section 2.5.2, RLL coding and some of its properties are presented. In section 2.5.3, some remarks are made concerning 2B1Q coding.

2.5.2 Run-length-limited coding

Run-length-limited (RLL) codes are often used in digital recording systems (e.g. Compact Disc) and sometimes in digital transmission systems because of their spectrum shape and presumed advantages for self clocking and reduced ISI. RLL codes impose a so called \((d, k)\) constraint on the (binary) output sequences of the encoder. A binary sequence satisfies the \((d, k)\) constraint if the number of consecutive zeros between adjacent ones is greater than or equal to \(d\) and less than or equal to \(k\).

The RLL sequence itself, denoted by \(\{b\}\), is not the input of the channel. The RLL sequence is used to generate the transmission signal \(W(t, \{\gamma\})\) (see fig. 2.8).

The sequence \(\{\alpha\}\) is a sequence of independent and equally distributed binary digits generated by the source. The digits are encoded into the RLL sequence, \(\{b\}\), by the RLL encoder. The (binary) modulation sequence, \(\{\gamma\}\), is formed from the RLL sequence \(\{b\}\). This process is defined by the equations

\[
c_n = c_{n-1} \oplus b_n, \tag{2.57}
\]

where \(\oplus\) denotes modulo two addition, and

\[
\gamma_n = 1 - 2c_n. \tag{2.58}
\]
The pulse train is formed using a zero-order hold (ZOH) device. Mathematically this is

\[ W(t, \{\gamma\}) = \sum_{n=-\infty}^{\infty} \gamma_n P_{\sigma}(t - n T_s), \]

where

\[ P_{\sigma}(t) = \begin{cases} 1, & 0 \leq t < T_s \\ 0, & \text{otherwise}. \end{cases} \]

To illustrate the process of transforming an RLL sequence into a modulation sequence, an example seems appropriate. The RLL sequence

\[ 0100101000 \ldots \]

would result in the modulation sequence (assuming \( c_{-1} = 0 \))

\[ +1 -1 -1 -1 +1 +1 -1 -1 -1 -1 \ldots \]

One may notice that the modulation sequence changes polarity iff the RLL sequence has a one.

Sequences \( \{b\} \) satisfying the \((d,k)\) constraint, may be thought of as being generated by a finite-state sequential machine (FSSM) as shown in figure 2.9. The probability that the FSSM is in state \( j \) is denoted by \( P(\sigma = j) \), where \( \sigma \) is a state variable. Formulas for \( P(\sigma = j) \) are derived later in this section.

If the FSSM is in state \( j \), where \( d \leq j < k \), then the next symbol generated may be a ‘0’ or a ‘1’ with corresponding next state \( j + 1 \) and 0 respectively. The probability that a ‘0’ is generated, given that the current state is \( j \), is denoted by \( p_j \). The probability that a ‘1’ is generated is denoted by \( q_j = 1 - p_j \). These transition probabilities are shown in figure 2.10.

From the FSSM for the generation of the RLL sequence \( \{b\} \), a FSSM that may be thought of as the generator of the modulation sequence \( \{\gamma\} \) is easily deducted (see fig. 2.11). The
upper $k + 1$ states can be labeled with: “The last generated symbol was a ‘1’.” For the lower $k + 1$ states a ‘0’ was generated last. Each state $j_u$ in the upper half of the diagram, has its corresponding state $j_l$ in the lower half and these states correspond to state $j$ in the diagram of figure 2.9. The probabilities that the FSSM, for the modulation sequence, is in state $j_u$ or $j_l$ are equal, and half as large as the probability that the FSSM for the RLL sequence is in state $j$

$$P(\sigma_w = j_u) = P(\sigma_w = j_l) = \frac{1}{2} P(\sigma = j), \quad (2.61)$$

where $\sigma_w$ is a state variable for the modulation sequence FSSM. The corresponding transition probabilities are equal:

$$p_{ju} = p_{jl} = p_j \quad (2.62)$$

and

$$q_{ju} = q_{jl} = q_j \quad (2.63)$$

To be able to state some of the properties of RLL codes, the code is examined in a different way. Any binary sequence can be parsed uniquely into a concatenation of phrases, each
phrase ending in a ‘1’ and beginning with zero, one or more ‘0’s. If the binary sequence satisfies the $(d, k)$ constraint, then the length of each phrase is at least $(d + 1)$ and no more than $(k + 1)$ digits. As an example, the sequence

$$0 1 0 1 0 0 1 0 1 1 0 1 \ldots$$

would parse as

$$0 1, 0 1, 0 0 0 1, 0 1, 1, 0 1, \ldots$$

where the commas separate the phrases.

Let $X_i$ be a random variable describing the number of binary digits in the $i^{th}$ phrase of the parsed sequence. Then the information rate of a random sequence of infinite length is defined as

$$R = \lim_{n \to \infty} \frac{H(X_1, X_2, \ldots, X_n)}{E(X_1, X_2, \ldots, X_n)} \quad (2.64)$$

Here $H(X_1, X_2, \ldots, X_n)$ is the joint entropy of the first $n$ phrases and $E(X_i)$ is the expectation of $X_i$.

**Maxentropic RLL codes**

A maxentropic RLL code is a $(d, k)$ code having maximal information rate. The maximum information rate of a code is denoted by the capacity $C$.

The properties of maxentropic RLL codes which are important in this work are now stated. For an explanation see Zehavi et al. [22]. It can be shown that, for a maxentropic RLL code, the variables $X_i$ are independent and identically distributed. The probability distribution of $X_i$ is

$$P(X_i = l) = 2^{-lC}, \quad l = d + 1, d + 2, \ldots, k + 1. \quad (2.65)$$

The capacity $C$ is the solution of

$$\sum_{l=d+1}^{k+1} 2^{-lC} = 1. \quad (2.66)$$

The average length of a phrase $X_i$ is
The probability that the FSM is in state \( j \) is

\[ P(\sigma = j) = P(\sigma = 0)P(X_i > j), \]  

where the state '0' probability is

\[ P(\sigma = 0) = 1/L. \]  

The transition probabilities \( p_j \) (fig. 2.10) are

\[ p_j = \frac{P(X_i = j + 1)}{1 - P(X_i \leq j)}. \]  

As before, \( q_j = 1 - p_j \).

In the remainder of this section, some practical RLL codes are discussed. These are (1,3), (1,7) and (2,7). From the work by Gallopoulos [9] and others it is known that, with respect to most properties, there is only one significant difference between a practical code and the corresponding maxentropic code with the same \((d,k)\) constraint. Namely, the analysis of practical codes is more involved, due to a larger number of states in the state transition diagram. Because other differences are small, the properties of maxentropic codes may be assumed to be a good approximation of the properties of practical codes.

Appendix A contains relevant data about the maxentropic (1,3), (1,7) and (2,7) RLL codes.

**Practical RLL codes**

Practical RLL codes operate at rates of the form \( m/n \), where \( m \) and \( n \) are small integers. Since maxentropic RLL codes have irrational rates, practical RLL codes are not optimal. Implementations of practical RLL codes discussed in this report are: Modified frequency modulation (MFM) \(^5\), which is a rate \( \frac{1}{2} \), (1,3) code, the Jacoby and the Adler-Hassner-Moussourie (AHM) codes, which are two implementations of a rate \( \frac{2}{3} \), (1,7) code and finally the IBM and Xerox implementations of a rate \( \frac{1}{2} \), (2,7) code.

Gallopoulos [9] has shown that Jacoby and the AHM coding produce output sequences with identical statistical behavior. The same is true for the IBM and Xerox rate \( \frac{1}{2} \), (2,7) RLL codes. The only difference lies in the rules that map the input stream into the output stream.

\(^{5}\)MFM is also known as “delay modulation” and as “Miller, code”
It was already stated that the FSMs corresponding to practical codes have more states than those for the corresponding maxentropic codes. As an example, in figure 2.12, the state transition diagram MFM is shown. See appendix A (and Howel [13]) for relevant data on the practical codes mentioned before.

In figure 2.12 all transition probabilities are equal to $\frac{1}{2}$ (except for the trivial probability one transitions).

2.5.3 2-Binary-to-1-Quaternary coding

The 2B1Q encoder is the concatenation of a serial-to-parallel converter with a 4-level mapper (fig. 2.13).

Note that this code has rate $R = 2$. 
1) The canceller 8 was derived from the canceller 7 by eliminating the 111 taps with the smallest values.

2) The canceller 9 contains a DC tap, 16 linear taps and all taps of order 2, 3 and 4 that combine components $a_{k-i}$ for $4 \leq i \leq 8$. The values 4 and 8 were chosen such that at least all important second and third order nonlinearities could be mimicked by the canceller.

### Table 2.3: Cancellers.

<table>
<thead>
<tr>
<th>canc</th>
<th>file 1</th>
<th>file 2</th>
<th>file 3</th>
<th>file 4</th>
<th>file 5</th>
<th>file 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Canc 1</td>
<td>30.51</td>
<td>35.93</td>
<td>27.89</td>
<td>36.15</td>
<td>33.95</td>
<td>33.57</td>
</tr>
<tr>
<td>Canc 2</td>
<td>30.46</td>
<td>35.91</td>
<td>27.54</td>
<td>35.91</td>
<td>33.96</td>
<td>33.61</td>
</tr>
<tr>
<td>Canc 3</td>
<td>30.54</td>
<td>35.95</td>
<td>27.87</td>
<td>36.04</td>
<td>34.00</td>
<td>33.59</td>
</tr>
<tr>
<td>Canc 4</td>
<td>30.56</td>
<td>37.20</td>
<td>27.81</td>
<td>36.73</td>
<td>35.13</td>
<td>35.12</td>
</tr>
<tr>
<td>Canc 5</td>
<td>30.67</td>
<td>40.59</td>
<td>27.41</td>
<td>36.50</td>
<td>37.48</td>
<td>37.30</td>
</tr>
<tr>
<td>Canc 6</td>
<td>30.20</td>
<td>35.94</td>
<td>27.39</td>
<td>36.34</td>
<td>33.56</td>
<td>33.34</td>
</tr>
<tr>
<td>Canc 7</td>
<td>29.50</td>
<td>45.40</td>
<td>26.29</td>
<td>42.39</td>
<td>36.58</td>
<td>40.25</td>
</tr>
<tr>
<td>Canc 8</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>40.61</td>
</tr>
<tr>
<td>Canc 9</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>40.24</td>
</tr>
</tbody>
</table>

### Table 2.4: Estimated SNR values in dB.
Chapter 3

Comparing the different schemes

3.1 Introduction

In chapter 2 the parts of "encoded CPM schemes" were discussed. In chapter 1 it was already stated that the assignment is to compare five different Gaussian Minimum Shift Keying (GMSK) schemes. How these schemes were compared is discussed in this chapter. The five schemes are:

- MFM (1,3) GMSK
- AHM (1,7) GMSK
- IBM (2,7) GMSK
- binary-uncoded GMSK
- 2B1Q GMSK,

where MFM stands for Modified Frequency Modulation, AHM stands for Adler-Hassner-Moussourie and 2B1Q stands for 2-Binary-to-1-Quaternary coding.

Note that, when a choice for one of the GMSK schemes is made, the transmission system is not completely determined. The energy per symbol $E$, the symbol duration time $T_s$ and the -3dB bandwidth $B_3$ of the Gaussian filter are still unspecified.

In the introduction of this report it was already mentioned that the channel is constrained by a function of the frequency, which denotes the maximum PSD at that frequency. This function limits the fraction of the transmitted power that ends up in the adjacent channels. This fraction is called the fractional out of band ratio. In practical systems, the restrictions are severe. As an example, K. Murota et al. [15] reports that the out of band power should generally be suppressed to 60-80 dB below the power in the desired channel in digital mobile radio applications.
Some facts which are true in general are now given without proof.

- Increasing the power of the Gaussian Minimum Shift Keying (GMSK) signal increases the baseband signal to noise ratio (SNR, see section 2.2) and thus the baseband channel capacity. It also increases the PSD of the CPM signal at all frequencies with a constant factor.

- Increasing the symbol rate on the channel, in general, increases the amount of information that is transmitted per unit of time (information rate) with the scheme. It also widens the PSD of the GMSK signal.

- Filtered pulse train signals $f_d(t, \gamma)$, with relatively small PSDs give rise to GMSK signals with relatively small PSDs.

- Increasing the amplitude of a filtered pulse train signal widens the PSD of the GMSK signal.

- Decreasing the -3dB bandwidth of the Gaussian filter narrows the PSD of the GMSK signal.

In the ideal case one would like to tune $E$, $T_s$ and $B_g$ for each modulation scheme, in such a way that the transmission scheme becomes optimal. In this work the optimal scheme is a scheme that meets the HF channel constraints and realizes the largest information rate. Theoretically it is possible to tune $E$, $T_s$ and $B_g$ to obtain the optimal scheme. One could make use of an iterative process. Start off with an arbitrary set $T_s$ and $B_g$, calculated the PSD of the CPM signal, then $E$ can be chosen as large as possible under the given constraint. Finally the information rate can be calculated. Then another point in the $T_s$-$B_g$ plane is chosen which results in a new value for the information rate. Proceeding uphill may lead to the optimal scheme. Unfortunately the calculation of the PSD of GMSK signals, generated with correlated input symbols, is very (computer) time consuming. Therefore a the following approach was chosen:

- For all modulation schemes, the binary sources that generate the equally distributed and independent data bits, have the same bit rate $1/T_b$ which is normalized to 1 bit/s. Thus the information rate is 1 bit/s.

- All modulation schemes use the same transmission power.

- Each scheme is evaluated for a limited number of values for the -3dB bandwidth of the Gaussian filter.

Since the different codes used have different entropy per symbol, the symbol rates on the channel must be adjusted accordingly (see table 3.1).

In summary, in this work different GMSK schemes are compared. For all schemes, the transmitted power and the information rate are identical. All schemes are evaluated for a
<table>
<thead>
<tr>
<th>code</th>
<th>entropy of symbol rate (bits/symbol)</th>
<th>symbol rate on the channel (symbols/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MFM (1,3)</td>
<td>1/2</td>
<td>2</td>
</tr>
<tr>
<td>AHM (1,7)</td>
<td>2/3</td>
<td>3/2</td>
</tr>
<tr>
<td>IBM (2,7)</td>
<td>1/2</td>
<td>2</td>
</tr>
<tr>
<td>binary uncoded</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2BIQ</td>
<td>2</td>
<td>1/2</td>
</tr>
</tbody>
</table>

Table 3.1: Entropy of channel symbols and symbol rates of the applied codes.

series of values for the Gaussian filter parameter \( B_g \). The GMSK schemes are judged on their baseband behaviour and the PSD of the GMSK signal. The judging of the baseband behaviour is discussed in section 3.2, and in section 3.3 the PSD calculation is discussed.

3.2 The eye pattern

To compare the baseband behavior of the schemes described in the introduction of this chapter, one could calculate the baseband channel capacity. The capacity depends on the baseband noise characteristics, and the wave forms that are selected to carry the channel symbols. For GMSK, a the input pulse train is filtered by a Gaussian filter. This determines the shape of the pulses that appear on the channel. The shape is only influenced by the symbol duration time \( T_s \), the symbol energy \( E \) and the -3dB down bandwidth of the Gaussian filter \( B_g \).

If \( f_1(t) \) and \( f_2(t) \) are two possible signal functions then the distance between these functions is defined as

\[
d_{12}\text{stackrel(\triangle)}=\int_{-\infty}^{\infty}|f_2(t)-f_1(t)|^2dt. \tag{3.1}
\]

The larger the distance between different wave forms the smaller bit error rate (BER) can be achieved. Another characteristic of a set of wave forms is the amount of intersymbol interference (ISI) it causes. If the ISI is large then detection is more difficult in general. To obtain a first impression of how difficult detection might be, the so called eye pattern gives much information (see appendix E). The eye pattern is obtained when shifts of the output

\[
f_d(t,\{\gamma\})=\pi h \sum_{n=-\infty}^{\infty} \gamma_n f(t-nT_s) \tag{3.2}
\]
of the Gaussian filter are written in the same graphic. The shifts need to be an integer number of times the symbol time \( T_s \). A good eye has two characteristics. The eye is wide open and it is wide. A wide open eye makes the discrimination between the different symbol levels more easy. If the eye is wide, then the timing of the sample moments is less critical.

### 3.3 HF spectrum calculation

It was already stated that in practical systems restrictions on the fractional out of band power are severe. To calculate (or estimate) the the fractional out of band power of a modulation scheme the PSD of the CPM signal must be determined first.

In the literature a large number of articles can be found that discuss the calculation of, and present the PSDs of various CPM signals. The reference list of one of these articles (see P.K.M. Ho [11]) contains the majority of them. Very few articles, however, contain PSDs of GMSK schemes, let alone coded GMSK schemes. For binary uncoded GMSK, PSDs are shown in the original article on GMSK by K. Murota et al. [15].

One of the articles found was an article by C. Chen et al., with the very promising title: "Spectral analysis of RLL codes modulated by FSK or CPM" [8]. However, the Continuous Phase Frequency Shift Keying (CPFSK) scheme described in this paper is totally different from the CPM schemes described in section 2.4 of this report. Chen divides the RLL sequence in phrases beginning with zero, one or more '0's and ending in a single ‘1’ (just as described in section 2.5.2). Then different frequencies are assigned to phrases with different lengths. The frequencies are chosen such that each phrase causes a phase change (modulo \( 2\pi \)) of exactly \( \pi \) radians. Furthermore, the polarity of the modulated signal is alternated phrase by phrase, which results in a continuous phase signal. The CPM schemes discussed in this report, show a phase deviation \( \pi h \gamma \) for each symbol that is transmitted. Here the phase deviation is proportional to the length of the phrase. In short, these are two completely different modulation schemes.

Since of most schemes discussed no PSDs were found in the literature, these had to be determined numerically. The calculation of the PSD of a CPM signal is not trivial, especially when the channel symbols are correlated and the frequency pulses are overlapping. Fortunately a large number of articles have been written on the subject. Almost all of the techniques described in these articles, fall into three categories. These are denoted by:

- Direct approach,
- Rowe-Prabhu chain approach,
- Autocorrelation function approach.

To be able to recognize a good approach when one is described, the following characteristics are desired (in order of importance). A good approach must be
1. able to handle correlated input symbols.

2. accurate enough to be used to calculate correct values for the fractional out of band power (which was already mentioned, should be suppressed in the order of 60-80 dB below the power in the desired channel). An accuracy of -100dB is desired which is very difficult to achieve.

3. of low computational complexity.

4. well described in the literature.

Note that in the literature, normally, not the PSD of the CPM signal itself is found. Instead, the PSD of the complex-valued signal

\[ V(t, \{\gamma\}) = \exp\{j(\phi(t, \{\gamma\}) + \phi_0)\} \]  \hspace{1cm} (3.3)

is determined. If the PSD of \( V(t, \{\gamma\}) \) is denoted by \( P_v(f) \) and assuming that the carrier frequency \( f_c \) is much greater than the bandwidth of \( V(t, \{\gamma\}) \), then, according to Prabhu and Rowe [16], the PSD of \( S(t, \{\gamma\}) \) is given by

\[ P_s(f) = \frac{E}{2T_s}[P_v(f - f_c) + P_v(-f - f_c)]. \]  \hspace{1cm} (3.4)

Furthermore, the \( P_v(f) \) is normally plotted in units of symbol rate on the channel \( (1/T_s) \). In this report, however, all PSD plots are normalized on the bit rate of the binary source \( (1/T_b) \), which is equal for all schemes. The advantage is that the plots can be compared without further scaling.

Now, the three approaches mentioned to calculate the PSDs are described briefly. The purpose is to show which approach best serves our needs.

**Direct approach**

The direct approach uses the fact that the PSD of a signal \( x(t) \) is given by

\[ P_x(f) = \lim_{T \to \infty} \frac{|X_T(f)|^2}{T}, \]  \hspace{1cm} (3.5)

where \( X_T(f) \) is the Fourier transform of the truncated version \( x_T(t) \) of \( x(t) \), defined by:

\[ x_T(t) = \begin{cases} x(t), & t < T/2, \\ 0, & \text{otherwise}. \end{cases} \]  \hspace{1cm} (3.6)
The complex-valued signal $V(t, \{\gamma\})$ is a stochastic signal. The different realizations of $\{\gamma\}$ correspond to different realizations of $V(t, \{\gamma\})$. The PSD of $V(t, \{\gamma\})$ is found by calculating the ensemble average

$$P_v(f) = \lim_{T_s \to \infty} \frac{1}{T_s} E \left[ |P_{v,T_s}(f, \{\gamma\})|^2 \right], \quad (3.7)$$

where $E[\cdot]$ denotes taking the ensemble average. The Fourier transform of the truncated version $V_{T_s}(t, \{\gamma\})$ of $V(t, \{\gamma\})$ is

$$P_{v,T_s}(f, \{\gamma\}) = \int_{-T_s/2}^{T_s/2} V_{T_s}(t, \{\gamma\}) \exp\{-j2\pi ft\} dt. \quad (3.8)$$

The idea is to evaluate equations 3.7 and 3.8 such that $P_v(f)$ is given as a function of simple properties of the frequency pulses used and the (statistical) properties of the generated channel symbol sequence $\{\gamma\}$. J. Salz [18] has evaluated the equations for a CPM scheme of which the output of the pulse shape filter can be represented as

$$f_d(t, \{\gamma\}) = \pi h \sum_{n=-\infty}^{\infty} \gamma_n r(t - nT_s), \quad (3.9)$$

where $r(t)$ is a rectangular pulse of length $T_s$

$$r(t) = \begin{cases} 1, & 0 \leq t < T_s \\ 0, & \text{elsewhere}. \end{cases} \quad (3.10)$$

The random variables $\gamma_n$ are assumed to be independent and identically distributed. The PSD is evaluated to a function of the expectation of the Fourier transform of the frequency pulse (with respect to the $\gamma_n$'s) and the characteristic function of the $\gamma_n$'s.

In another paper by J.E. Mazo and J. Salz [14], the output of the pulse shape filter, is allowed to be more general

$$f_d(t) = \sum_{n=-\infty}^{\infty} s_n(t - nT_s). \quad (3.11)$$

Here the $s_n(t)$ are independent and identically distributed stochastic signals defined on the interval $[0, T_s]$. The same calculations are performed and the same sort of formula is found.

The direct approach has the following disadvantages:

- No extension evaluation of the approach to correlated data symbols was found.
• Even worse, extension to overlapping frequency pulses was found.

• According to P.K.M. Ho et al. [11], integration in one or two dimensions is usually necessary, which, again according to Ho, may lead to inaccurate results at frequencies far from the carrier frequency.

The direct approach does not seem to have any advantages for our application.

**Rowe-Prabhu approach**

The Rowe-Prabhu approach was first described by V.K. Prabhu and H.E. Rowe [16]. In this article the aim is to calculate the PSD of phase shift keying (PSK) signals of the form

\[
S(t, \{\gamma\}) = \sqrt{\frac{2E}{T_s}} \cos(2\pi f_c t + \phi(t, \{\gamma\}) + \phi_0),
\]

where

\[
\phi(t, \{\gamma\}) = \sum_{n=\infty}^\infty g_{n\gamma}(t - nT_s), \quad \gamma = 1, 2, \ldots, M.
\]

As before, \( E \) is the energy per symbol, \( T_s \) is the symbol duration time, \( f_c \) is the carrier frequency, and \( \phi_0 \) is the phase deviation at time \( t = 0 \). Note that for each signaling interval one of \( M \) different time functions is transmitted.

Instead of the PSD of \( S(t, \{\gamma\}) \), the PSD of \( V(t, \{\gamma\}) \) is calculated (as explained before). Here \( V(t, \{\gamma\}) \) is given by equation 3.3 and \( \phi(t, \{\gamma\}) \) is given by equation 3.13.

The main step in the Rowe-Prabhu approach is to represent the complex-valued signal \( V(t, \{\gamma\}) \) as a baseband pulse train:\(^1\)

\[
V(t, \{\gamma\}) = \sum_{n=\infty}^\infty b_n r(t - nT_s)],
\]

where \( r(t) ] \) satisfies

\[
r(t) ] = 0], \quad t < 0] \lor t \geq T_s,
\]

and each \( b_n \) is a unit basis vector. The entries of \( r(t) ] \) are all possible shapes that \( V(t, \{\gamma\}) \) can have in an arbitrary time slot. The vector \( b_n \) selects one of the pulse shapes for time slot \( \gamma \).

\(^1\)Row and column vector variables are represented as \( \mathbf{a} \) and \( \mathbf{a}^\top \), respectively, matrices are represented as \( [\mathbf{A}] \).
This representation is possible if the time functions \( g_i(t) \) are limited to an interval \( LT_s \). In that case, during an arbitrary time slot, there are only \( L \) time functions contributing the phase deviation, \( \phi(t, \{ \gamma \}) \) (see equation 3.13). And, since each time function is chosen from a set of \( M \) functions, the phase deviation has one of only \( M^L \) shapes. The complex signal \( V(t, \{ \gamma \}) \) is a function of the phase (as defined in equation 3.3). Therefore, \( V(t, \{ \gamma \}) \) also has one of only \( M^L \) different shapes. Note that the sequence \( \{ h \} \) is a function of the sequence \( \{ \gamma \} \) only. The elements of \( r(t) \) are determined by the set of signaling time functions \( g_i(t) \).

For signals of the form of equation 3.14, Prabhu and Rowe [16] found that the PSD equals\(^2\)

\[
P_v(f) = R(f)[P_b(fT_s)]R^*(f),
\]

(3.16)

where the elements of \( R(f) \) are the Fourier transforms of the elements of \( r(t) \) and \( P_b(f, T_s) \) is the Fourier sum of the autocorrelation matrix of the sequence \( \{ b \} \) defined by

\[
[P_b(fT_s)] = \sum_{n=-\infty}^{\infty} [\Phi_b(n)]e^{-j2\pi fT_sn},
\]

(3.17)

where

\[
[\Phi_b(n)] = E[ b_{k+n}d_k^* ].
\]

(3.18)

The only assumption made while deriving equation 3.16 was that the \( b_k \) are wide-sense stationary. Note that the vectors and matrices are of size \( M^L \).

In a second article, H.E. Rowe and V.K. Prabhu [17] extended the their method to cover the case of phase shift keying (PSK). Note that a CPM signal is a special form of a PSK signal. Again, it is assumed that the baseband signaling pulses are limited to an interval \( LT_s \).

The major difference between an FSK and signal and a PSK signal is that for an FSK signal the current phase depends on all previous transmitted baseband signaling pulses; for PSK, on the other hand, it was already stated that only \( L \) time functions contribute to the phase in a particular time slot. However, the situation is not as bad as it looks. It is not difficult to see that the contribution of all but \( L \) base band signaling pulses to the phase in a particular time slot the addition of a constant phase deviation in that time slot (see for example equation 2.46 and 2.47 for the CPM case). Thus, except for this constant phase deviation, the phase in each time slot again has one of \( M^L \) shapes. Rowe and Prabhu [17] showed that also for FSK signals it is possible to find an expression for the complex signal \( V(t, \{ \gamma \}) \) of the form

---

\(^2\)The token \(^*\) denotes complex conjugation.
\[ V(t, \{\gamma\}) = \sum_{n=-\infty}^{\infty} \xi_n r(t - nT_s) \],

(3.19)

where the elements of \( r(t) \) are formed by the \( M^L \) different shapes of \( V(t, \{\gamma\}) \) during one time slot, not taking into account the influence of the constant phase deviation discussed above. The sequence \( \{\xi_n\} \) selects, for each time slot, one of the pulse shapes in \( r(t) \). Here, the one non-zero element of each \( \xi_n \) takes care of the contribution of the constant phase deviation.

Since the FSK signal now is presented in the same form as the PSK signal, the PSD of the FSK signal is given by equation 3.16 with \( b \) replaced by \( c \):

\[ P_c(f) = R(\hat{f})[P_c(fT_s)]R^*(f) \],

(3.20)

where \( R(\hat{f}) \) is as explained before and \([P_c(fT_s)]\) is given by

\[ [P_c(fT_s)] = \sum_{n=-\infty}^{\infty} [\phi_c(n)]e^{-j2\pi fT_sn}, \]

(3.21)

where

\[ [\phi_c(n)] = E[ c_{k+n}\xi_n^*]. \]

(3.22)

For the calculation of PSDs of FSK signals Rowe and Prabhu [17] made the assumption that the symbol sequence \( \{\gamma\} \) consists of independent symbols. Note that the size of the vectors and matrices is \( M^L \).

P.K.M. Ho and P.J. McLane extended the Rowe-Prabhu method for the calculation of the PSD of CPM signals with a correlated data input sequence \( \{\gamma\} \) [12]. In this article Ho and McLane assume that the input sequence may be thought of as being generated by a finite state sequential machine (FSSM). If the number of states of the FSSM is \( N \) and the number of one-step code transitions from each state is \( K \) then the total number of one-step code transitions is \( Q = NK \). The most general scheme would transmit a different channel symbol for each different one-step transition of the FSSM. Thus there are \( Q \) different signaling functions of the form \( \pi b\gamma_n f(t) \) (see equation 2.46 and 2.47 on page 27). If the signaling functions could be transmitted independently and assuming that the pulse \( f(t) \) has length \( LT_s \), then the vector function \( r(t) \) would contain \( Q^L \) different pulse shapes. However, the signaling functions are not independent. Assume that in the \( n^{th} \) time slot the FSSM is in state \( i \), then there are \( K \) one-step transitions possible, corresponding to \( K \) signaling functions. If the reached state is \( j \) then again there are \( K \) one-step transitions possible, etc. Thus if the length of \( f(t) \) is \( LT_s \) then \( L \) consecutive signaling functions contribute to the shape in a single time slot. For the generation of this sequence the FSSM can start in one of \( N \) states and for each of the \( L \) transitions there are
$K$ alternatives. This means that $r(t)$ contains $NK^L$ different pulse shapes. Using this $r(t)$ the CPM signal can be represented as the base band pulse train of equation 3.19, and the PSD is again given by equations 3.20, 3.21 and 3.22. Rowe and Prabhu [17] and Ho and McLane [12] showed that, when the PSD does not contain spectral lines, then the geometric matrix series in equation 3.21 can be written as:

$$\sum_{n=-\infty}^{\infty} [\phi_c(n)] \exp{-j2\pi fT_s n} = [A] + [A]^\dagger,$$

(3.23)

where $[A]$ is a calculated at the cost of a few matrix multiplications and additions (see Ho and McLane [12]).

Important here is that the complexity of the calculations of the PSD grows with the size of the vectors and matrices, which is $NK^L$. Thus the complexity is exponential in the length of the signaling function $f(t)$. This means that the complexity grows very fast for longer signaling pulse functions.

The Rowe-Prabhu approach has the following advantages:

- A description of the approach for the calculation of the PSD of CPM signals with correlated data symbols and overlapping frequency pulses is available (see P.K.M. Ho and P.J. McLane [12]).
- Except for the calculation of $R(f)$, no integration is necessary. According to Ho and McLane [12] this means that the accuracy of the PSD at frequencies far from the carrier frequency may be expected to be higher with this method than with the other two methods.

The Rowe-Prabhu approach also has one disadvantage:

- The complexity of this method is proportional to $NK^L$ (with $N$ the number of states of the FSSM, $K$ the number of one-step transitions in each state and $LT_s$ the length of the frequency pulse $f(t)$).

**Autocorrelation function method**

It is well known that the PSD of a signal can be calculated by first calculating its autocorrelation function, and subsequently calculating the Fourier transform of this autocorrelation function. The autocorrelation function of the complex signal $V(t, \{r\})$ defined in equation 3.3, is

$$R(t, \tau) = E[V(t + \tau, \{r\})^* V(t, \{r\})],$$

(3.24)

$^3$The token $^\dagger$ denotes taking the Hermitian transpose.
where $E[·]$ denotes taking the ensemble average over all sequences $\gamma$. The PSD of $V(t, \gamma)$ is

$$P_v(f) = \int_{-\infty}^{\infty} R(\tau) \exp\{-j2\pi f \tau\} d\tau,$$

where $R(\tau)$ is the average autocorrelation function given by

$$R(\tau) = \frac{1}{T_s} \int_0^{T_s} R(t, \tau) dt.$$  \hspace{1cm} (3.26)

Equation 3.26 is valid when $R(t, \tau)$ is cyclostationary with period $T_s$. For this to be true it is sufficient to show that $\gamma_n$ is stationary.

There are many articles in literature that cover the autocorrelation function approach. Especially T. Aulin and C.-E.W. Sundberg have investigated the subject thoroughly for the case of CPM signals [1,2,3,4]. Most articles, however, do not allow the input data symbols to be correlated. One of the most recent articles that discusses the autocorrelation function approach is part one of the article by P.K.M. Ho and P.J. McLane [11]. In this article Ho and McLane present an implementation of the approach which is especially adapted to calculate the PSD of a general CPM signal generated using correlated input data symbols. They show that the process $\gamma_n$ is stationary (see appendix D). It was already stated that, in that case, equation 3.26 is valid. Combining equations 3.25 and 3.26 and using the fact that $R(\tau)$ is an even function yields

$$P_v(f) = \frac{2}{T_s} \text{Re} \left[ \int_{t=0}^{T_s} \int_{\tau=0}^{\infty} R(t, \tau) \exp\{-j2\pi f \tau\} d\tau dt \right],$$

where $\text{Re}[x]$ denotes taking the real part of the complex number $x$.

Use of the formula for $\phi(t, \gamma)$ (equation 2.46) in the formula for $V(t, \gamma)$ (equation 3.3) and use of the result in the equation 3.24 yields

$$R(t, \tau) = E \left[ \prod_{n=-L+1}^{m} \exp\{j2\pi h \gamma_n, p(t-nT_s, \tau)\} \right],$$

$$0 \leq t < T_s, mT_s \leq t + \tau < (m+1)T_s, \tau \geq 0,$$

where

$$p(t, \tau) = g(t + \tau) - g(t).$$

How these equations are derived is explained in more detail in appendix D. In the equation 3.28 it is used that $p(t)$ is zero outside the interval $0 < t < LT_s$ and the point $(t, \tau)$ is in
the limited area. Note that the averaging must be carried out over $M^{L+m}$ sequences. The next step Ho and McLane made was to introduce the function $\psi^i_m(t, \tau)$. This function represents the autocorrelation of $V(t, \{\gamma\})$, given by equation 3.28, assuming that the sequence of states involved in equation 3.28 starts with state $\sigma_i$. The autocorrelation function can then be rewritten as

$$R(t, \tau) = \sum_{i=1}^{N} p_i \psi^i_m(t, \tau) = P \Psi_m(t, \tau),$$

$$0 \leq t \leq T_s, mT_s \leq t + \tau < (m + 1)T_s,$$  \hspace{1cm} (3.30)

where

$$\Psi_m(t, \tau) = [\psi^1_m(t, \tau), \psi^2_m(t, \tau), \ldots, \psi^N_m(t, \tau)]$$  \hspace{1cm} (3.31)

and $P = [p_1, p_2, \ldots, p_N]$ of which the elements $p_i$ denote the probability that the encoder is in state $\sigma_i$. After some mathematical manipulation, Ho and McLane managed to derive the following recursive equation for $\psi^i_m(t, \tau)$:

$$\psi^i_{m+1}(t, \tau) = \sum_j \left\{ \sum_{k=1}^{N_p} t_{ik} \exp\{j\pi h S_{ij}^k(t + (L - 1)T_s, \tau)\} \right\} \psi^j_m(t - T_s, \tau),$$  \hspace{1cm} (3.32)

where $t_{ij}^k$ is the probability that, given state $\sigma_i$ the encoder goes to the next state $\sigma_j$ via the one-step transition $k$. During this transition the channel symbol $S_{ij}^k$ is emitted. Equation 3.32 can be rewritten in matrix form

$$\Psi_{m+1}(t, \tau) = [X(t + (L - 1)T_s, \tau)] \Psi_m(t - T_s, \tau),$$  \hspace{1cm} (3.33)

where the elements of $[X(t, \tau)]$ are simple functions of the phase response $g(t)$ and the channel symbols $S_{ij}^k$. Note that all vectors and matrices are of size $N$. Evaluating the recursion yields:

$$\Psi_{m+1}(t, \tau) = \left( \prod_{n=1-L}^{m-L} [X(t - nT_s, \tau)] \right) \Psi_0(t - mT, \tau).$$  \hspace{1cm} (3.34)

It can be shown that the matrix product in equation 3.34 is independent of $\tau$, furthermore $\Psi_0(t - mT, \tau)$ can be rewritten in terms of matrices $X(t, \tau)$. This leads to a new expression for the autocorrelation function

$$R(t, \tau) = P \left( \prod_{n=1-L}^{m-L} [Z(t - nT_s)] \right) \left( \prod_{n=-L+1}^{0} [X(t - nT_s, \tau)] \right) 1,$$  \hspace{1cm} (3.35)
where \([Z(t)]\) is a simplified version of \([X(t, \tau)]\) using the fact that length of the baseband signaling pulse and the domain of \((t, \tau)\) are limited and \([1, 1, \ldots, 1]^T\). Note that the complexity of equation 3.35 is linear in the length of the baseband pulse. Of course, the complexity of the equation is also linear in \(m\). And, since the integration in equation 3.27 extends over an area where \(m\) goes to infinity, the complexity of calculating the PSD, using equation 3.35, would be infinite. Fortunately for larger values of \(m\), equation 3.35 can be simplified further. To this end, Ho and McLane divided the area of integration into two parts. The first part is the area where \(t + \tau < LT_s \) \((m < L)\) and the second part is the area where \(t + \tau \geq LT_s \) \((m \geq L)\). The results of the integration over these two areas is added in the end. The the integral of the first part is denoted by \(P_1(f)\) and is defined by

\[
P_1(f) = \int_{t=0}^{T_s} \int_{\tau=0}^{LT_s-t} R(t, \tau) \exp \{-j2\pi f \tau\} d\tau dt,
\]

where the autocorrelation function is calculated using the formula in equation 3.35. When \(m \geq L\) then the formula for \(R(t, \tau)\) can be simplified further. Ho and McLane found the expression of the form

\[
R(t, \tau) = E[H(t)][W]^{m-L} A(t + \tau - nT_s),
\]

\(m \geq L, \ 0 \leq t < T_s, \ mL \leq t + \tau < (m + 1)T_s,\)

where \([H(t)]\) is a product of \(L \ [Z(t)]\)-matrices, \([W]\) is a constant matrix and \([A(t)]\) is a product of \(L \ [X(t, \tau)]\)-matrices. Integration and simplification yields for the integral of the second part

\[
P_2(f) = \frac{1}{T_s} \exp \{-j2\pi f LT_s\} E[H(f)] \left( \sum_{n=0}^{\infty} [W]^n \exp \{-j2\pi n f T_s\} \right) A(f),
\]

where

\[
A(f) = \int_{0}^{T_s} A(\tau) \exp \{-j2\pi f \tau\} d\tau,
\]

and

\[
[H(f)] = \int_{0}^{T_s} [H(t)] \exp(\pm j2\pi f t) dt.
\]

In case when the series converges, and thus no line components are present, we have

\[
R(t, \tau) = \frac{1}{T_s} \exp \{-j2\pi f LT_s\} E[H(f)] [[I] - \exp \{-j2\pi f T_s\}[W]^{-1}} A(f).
\]

\(^{4}\)The token \(T\) denotes taking the transpose.
Finally the PSD of $V(t, \gamma)$ is

$$P_v(f) = 2 \cdot \text{Re}[P_1(f) + P_2(f)]. \tag{3.42}$$

Drawbacks:

- Integration in two dimensions is necessary to calculate the average autocorrelation function $R(\tau)$. This means that also for this technique inaccurate results at frequencies far from the carrier frequency may be expected.

The autocorrelation function method has also advantages:

- A good description of this technique allowing correlated input symbols is found in an article by P.K.M. Ho and P.J. McLane [11].

- Computational complexity is linear in the pulse length $L$.

In this work the autocorrelation function technique was chosen. The reason for this is explained now. The direct approach was rejected because it has the already mentioned drawbacks and it has no advantages for our application, to compensate for them. The Rowe-Prabhu and the autocorrelation function method both have advantages and disadvantages. Which of the techniques is to be preferred depends on the exact form of the premodulation encoder used. Ho [12] advises to use the autocorrelation function method if for the encoder $NK^L > 64$, here $N$ is the number of states in the accompanying FSSM, $K$ is the number of transitions at each state and $L$ is the length of the frequency pulse in symbol times. For the IBM (2,7) run length limited code it is shown in appendix C that $N = 24$ and $K = 2$. Theoretically the GMSK frequency pulse is of infinite length. To be able to calculate psds the pulse must be truncated to an interval of length $LT_x$ such that the pulse is sufficiently small outside the interval. Calculations have shown that for small values of the GMSK filter parameter $BT_x$, $L$ must be ten or more symbol times. For this example $NK^L = 24 \cdot 2^{10} \gg 64$ thus for this example the autocorrelation function method must be chosen. Note that, even though the Rowe-Prabhu method may be more efficient and/or more accurate for small values of $N$ and $L$, the PSD can still be calculated using the autocorrelation function method. If the calculation of the PSD of these schemes with small values of $N$ and $L$ is compared with the calculation of the more complex schemes. Then it is safe to assume that the computational complexity of the first one is far smaller than that of the second. And there is no reason to expect the accuracy of the first being worse compared that of the second.

In literature, with the autocorrelation function method an accuracy $\pm 90 \text{dB}$ is achieved. This can be improved upon by using double precision arithmetic. It was found that an accuracy of $\pm 120 \text{ dB}$ may be achieved easily in this way.
Chapter 4

Results

First, it has to be explained more precise, how the different schemes were compared. Remember, the approach introduced in chapter 3:

- For all modulation schemes, the binary sources that generate the equally distributed and independent data bits, have the same bit rate $1/T_b$ which is normalized to 1 bit/s. Thus the information rate is 1 bit/s.

- All modulation schemes use the same transmission power.

- Each scheme is evaluated for a limited number of values for the -3dB bandwidth of the Gaussian filter.

Item three says that the each scheme is evaluated for a limited number of values for the -3dB bandwidth of the Gaussian filter. Of course, for the comparison to be fair, the same set of Gaussian filters must be used for all schemes. Therefore the -3dB bandwidth of the Gaussian filter is normalized on the bit rate, $1/T_b$, of the binary source. The formulas in this report all contains the symbol rate $1/T_s$ instead of the bit rate (e.g. see equations 3.2 and 2.55, which are used ,to calculate the eye-pattern). As an example, lets compare uncoded GMSK and (1,3) RLL GMSK. For uncoded GMSK $\frac{1}{T_s} = \frac{1}{T_b}$, for (1,3) RLL coded GMSK $\frac{1}{T_s} = \frac{1}{T_b}$. Thus if calculations with $B_g T_b = 0.2$ need to be performed, $B_g T_s = B_g T_b = 0.2$ for the uncoded case, but $B_g T_s = \frac{1}{2} B_g T_b = 0.1$. In table 4.1 the values for $B_g T_s$ corresponding to the values 0.2, 0.25, 0.5 and 1 for $B_g T_b$ are gathered.

The results obtained, using the formulas in this report are normalized the the symbol rate. To be able to compare the results of different schemes the results must be normalized on the bit rate. In case of the eye-pattern calculations, we chose to depict two bit times $T_b$. As a result the eye-pattern belonging to the uncoded GMSK system has two eyes. For the (1,3) RLL GMSK system, however, $T_s = \frac{1}{2} T_b$, thus four eyes appear. The eye-pattern of the 2B1Q GMSK system has only one eye.
Table 4.1: Values for $B_g T_s$

<table>
<thead>
<tr>
<th>code</th>
<th>symbol rate on channel (symbols/s)</th>
<th>$B_g T_s$</th>
</tr>
</thead>
<tbody>
<tr>
<td>MFM</td>
<td>2</td>
<td>0.1 0.125 0.25 0.5</td>
</tr>
<tr>
<td>IBM(1,7)</td>
<td>3/2</td>
<td>0.133 0.166 0.333 0.667</td>
</tr>
<tr>
<td>IBM(2,7)</td>
<td>2</td>
<td>0.1 0.125 0.25 0.5</td>
</tr>
<tr>
<td>2-level uncoded</td>
<td>1</td>
<td>0.2 0.25 0.5 1</td>
</tr>
<tr>
<td>4-level uncoded</td>
<td>1/2</td>
<td>0.4 0.5 1.0 2.0</td>
</tr>
</tbody>
</table>

Table 4.2: Multiplication factors

<table>
<thead>
<tr>
<th>code</th>
<th>symbol rate on channel (symbols/s)</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>MFM</td>
<td>2</td>
<td>1/2</td>
<td>2</td>
<td>1/2</td>
</tr>
<tr>
<td>IBM(1,7)</td>
<td>3/2</td>
<td>2/3</td>
<td>3/2</td>
<td>2/3</td>
</tr>
<tr>
<td>IBM(2,7)</td>
<td>2</td>
<td>1/2</td>
<td>2</td>
<td>1/2</td>
</tr>
<tr>
<td>2-level uncoded</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4-level uncoded</td>
<td>1/2</td>
<td>2</td>
<td>1/2</td>
<td>2</td>
</tr>
</tbody>
</table>

1. Multiplication factor for the vertical axis (eye-pattern).

2. Multiplication factor for the vertical axis (PSD).

3. Multiplication factor for the horizontal axis (PSD).

The calculated PSDs are also normalized on the symbol rate. We chose to calculate the PSD in the range $0 \leq fT_b < 5$. What must be done to transform a PSD, normalized on the symbol rate $1/T_s$ to a PSD normalized on the normalized on the bit rate $1/T_b$. For the (1,3) RLL GMSK code, if $fT_s = 5$ then $1/2 fT_b = 5$, or $fT_b = 10$. Thus the PSD must be multiplied against the vertical axis with a factor 2. But there is a snake in the grass! Multiplying against the vertical axis with factor two also doubles the transmitted energy. To compensate for this, all calculated samples of the PSD must divided by 2. For all schemes the multiplication factors are presented in table 4.2.

The calculated eye-patterns and PSDs can be found in appendix E.

How accurate are the PSD calculations? The accuracy of the PSDs was determined in two ways. First the PSD of a minimum shift keying signal was calculated. Since there
excists a closed form formula for this PSD, values of the PSD can be calculated exact. The calculated PSD was accurate to about 140 dB. From the PSDs in appendix E we see that the PSD becomes inaccurate at about 120 dB.
Chapter 5

Conclusion

A lot of time was spend to realize the accurate PSD estimation of encoded GMSK signals. The conclusions however can be short.

Comparing the eye-patterns of 2B1Q GMSK with the others shows that timing of the sample moments is less critical. For $B_s T_b = 0.25$, the eye-pattern is much better than the eye-patterns of the other schemes. If one compares the PSDs of 2B1Q with the PSDs of the other schemes, then it is observed that all calculated PSDs of 2B1Q outperform the PSDs of the other schemes.

Although the symbol rate for, for example (1,3) RLL GMSK is twice the symbol rate of uncoded GMSK the PSDs of (1,3) RLL GMSK aren’t twice as wide as the PSDs of uncoded GMSK. This supports our expectation that encoding of the input data stream narrows the PSD.

J.P.J. Kemmelings
Eindhoven, February 9, 1993
Philips Electronics N.V.
Bibliography


Appendix A

Characterization results, file 1

A.1 Impulse response
A.2 Noise PSD
A.3 Amplitude response
Appendix B

Gaussian frequency pulses
Appendix C

RLL data

This appendix contains all relevant data about the RLL codes used in this report. The following information can be found:

- $R$, rate of the code
- $p_i$, probability for a transition from state $i$ to state 0
- $q_i$, probability for a transition from state $i$ to state $i+1$
- $P(u = i)$, probability for the encoder to reside in state $i$
- constraint graphs for practical codes

The RLL codes to be found are:

- (1,3) optimal
- (1,7) optimal
- (2,7) optimal
- (1,3) MFM
- (1,7) IBM
- (2,7) IBM

The practical codes have more constraints than just the $(d, k)$ constraint. These constraints are reflected in the constraints graphs for these codes.

All data has an accuracy of 15 digits.
Table C.1: Data optimal (1,3) RLL code

<table>
<thead>
<tr>
<th>$R$</th>
<th>0.55146 30897 45595</th>
<th>$P(u=0)$</th>
<th>0.36347 96919 83575</th>
<th>$P(u'=0)$</th>
<th>0.18173 98459 91787</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p_1$</td>
<td>0.46557 12318 76768</td>
<td>$q_1$</td>
<td>0.53442 87681 23231</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$p_2$</td>
<td>0.59441 44760 16249</td>
<td>$q_2$</td>
<td>0.40558 55239 83750</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$P(u=1)$</td>
<td>0.36347 96919 83575</td>
<td>$P(u'=1)$</td>
<td>0.18173 98459 91787</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$P(u=2)$</td>
<td>0.19425 40040 24594</td>
<td>$P(u'=2)$</td>
<td>0.09712 70020 12296</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$P(u=3)$</td>
<td>0.07878 66120 08256</td>
<td>$P(u'=3)$</td>
<td>0.03939 33060 04127</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table C.2: Data optimal (1,7) RLL code

<table>
<thead>
<tr>
<th>$R$</th>
<th>0.67928 62637 47264</th>
<th>$P(u=0)$</th>
<th>0.29466 15372 43914</th>
<th>$P(u'=0)$</th>
<th>0.14733 07686 21957</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p_1$</td>
<td>0.38996 79522 24187</td>
<td>$q_1$</td>
<td>0.61003 20477 75812</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$p_2$</td>
<td>0.39920 01775 93620</td>
<td>$q_2$</td>
<td>0.60079 98224 06379</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$p_3$</td>
<td>0.41493 05284 65252</td>
<td>$q_3$</td>
<td>0.58506 94715 34747</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$p_4$</td>
<td>0.44287 62698 46084</td>
<td>$q_4$</td>
<td>0.55712 37301 53915</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$p_5$</td>
<td>0.49641 53616 44604</td>
<td>$q_5$</td>
<td>0.50358 46383 55395</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$p_6$</td>
<td>0.61558 38218 86648</td>
<td>$q_6$</td>
<td>0.38441 61781 13351</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$P(u=1)$</td>
<td>0.29466 15372 43914</td>
<td>$P(u'=1)$</td>
<td>0.14733 07686 21957</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$P(u=2)$</td>
<td>0.17975 29809 65674</td>
<td>$P(u'=2)$</td>
<td>0.08987 64904 82837</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$P(u=3)$</td>
<td>0.10799 55590 41194</td>
<td>$P(u'=3)$</td>
<td>0.05399 77795 20597</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$P(u=4)$</td>
<td>0.06318 49046 56331</td>
<td>$P(u'=4)$</td>
<td>0.03159 24523 28165</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$P(u=5)$</td>
<td>0.03520 18097 71554</td>
<td>$P(u'=5)$</td>
<td>0.01760 09048 85777</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$P(u=6)$</td>
<td>0.01772 70906 43263</td>
<td>$P(u'=6)$</td>
<td>0.00886 35453 21631</td>
<td></td>
<td></td>
</tr>
<tr>
<td>$P(u=7)$</td>
<td>0.00681 45804 34152</td>
<td>$P(u'=7)$</td>
<td>0.00340 72902 17076</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure C.1: FSM for MFM
Table C.3: Data optimal (2,7) RLL code

<table>
<thead>
<tr>
<th>code</th>
<th>R</th>
</tr>
</thead>
<tbody>
<tr>
<td>MFM (1,3)</td>
<td>1/2</td>
</tr>
<tr>
<td>AHM (1,7)</td>
<td>2/3</td>
</tr>
<tr>
<td>IBM (2,7)</td>
<td>1/2</td>
</tr>
</tbody>
</table>

Figure C.2: Constraint graph for the IBM (1,7) code
Figure C.3: Constraint graph for the IBM (2,7) code
Appendix D

Autocorrelation function method

D.1 Introduction

In this appendix the autocorrelation function method is described in detail. The description is based on the article by P.K.M. Ho and P.J. McLane [11]. The parts of the article that are already discussed in detail in the main part of this report are omitted (e.g. the description of a CPM system, only the important formulas are repeated). Some parts of the article are very concise, they are discussed in more detail here.

In this appendix, for the simplicity of the notation, the parameter \( \{ \gamma \} \) is omitted in \( S(t,\{ \gamma \}) \), \( V(t,\{ \gamma \}) \) and \( \phi(t,\{ \gamma \}) \).

D.2 The encoder

P.K.M. Ho and P.J. McLane [11] assume that the channel symbol sequence \( \{ \gamma \} \) may be thought of as being generated by an encoder fed with independent and identically distributed symbols \( \{ a \} \) from a binary source. The result of this assumption is that the encoder and the binary source may be modeled as a finite-state sequential machine (FSSM) of the Mealey type. This type of encoder is characterised by a set of states \( \{ 0^1, 0^2, \ldots, 0^N \} \) with specified state transitions and by a set of output symbols \( \{ \xi_1, \xi_2, \ldots, \xi_M \} \).

Now some important concepts concerning FSSMs are introduced. Let \( u_i \) and \( \gamma_i \) be the state and the output of the encoder at time \( IT_s \), where \( u_i \in \{ \sigma_1, \sigma_2, \ldots, \sigma_N \} \) and \( \gamma_i \in \{ \xi_1, \xi_2, \ldots, \xi_M \} \). The transition probability \( \phi_{ij} \) is the probability that the encoder changes to state \( \sigma_j \) when the current state is \( \sigma_i \), i.e.

\[
\phi_{ij} = P(u_{i+1} = \sigma_j \mid u_i = \sigma_i). \tag{D.1}
\]

The transition probability \( \phi_{ij} \) is the \((i,j)^{th}\) element of the state transition matrix [\( \Phi \)].
Note that

\[ \sum_{j=1}^{N} \phi_{ij} = 1 \quad \forall i. \]  \hspace{1cm} (D.2)

Ho and McLane allow parallel transitions between two states. The number of parallel transitions between two different states is assumed to be \( N_p \). The fact that \( N_p \) is the same number for all transitions is no restriction. If necessary one can add extra transitions with probability of occurrence equal to zero. If a transition from state \( \sigma_i \) to state \( \sigma_j \) occurs in the \( k^{th} \) way then the output of the encoder is \( S_{ij}^k \), where \( S_{ij}^k \in \{\xi_1, \xi_2, \ldots, \xi_N\} \). The probability that \( S_{ij}^k \) is transmitted, given that the current state is \( \sigma_i \), is \( t_{ij}^k \). Mathematically \( t_{ij}^k \) is defined as

\[ t_{ij}^k = P(\gamma_l = S_{ij}^k | u_l = \sigma_i), \]  \hspace{1cm} (D.3)

and is called the symbol probability. Of course the probability that one of the \( N_p \) ways to change to state \( \sigma_j \), given current state \( \sigma_i \), is taken equals the transition probability \( \phi_{ij} \):

\[ \phi_{ij} = \sum_{k=1}^{N_p} t_{ij}^k. \]  \hspace{1cm} (D.4)

Figure D.1 shows a state transition diagram of a FSSM with two states. Note that \( N_p = 2 \).

Ho and McLane assume that the set of \( N \) states of the encoder form an irreducible set. A set of states is said to be irreducible if, from each state in the set, no state outside the set can be reached and every state within the set can be reached in one or more steps. Such an assumption is valid for all encoders of practical interest, including those discussed in this report. The period of an irreducible set of states is the largest integer \( J \) such that all possible recurrence times for states are a multiple of \( J \). If \( J > 1 \), the irreducible set of states is called periodic, as in figure D.1, where \( J = 2 \). If \( J = 1 \), the irreducible set is called ergodic. The autocorrelation function approach of Ho and McLane can be used for both the periodic and the ergodic case.
In the following, use is made of the initial state probabilities. Ho and McLane assumed that the encoder is started in the infinite past at state $\sigma_i$ with probability $p_i$, the vector

$$P = [p_1, p_2, \ldots, p_N] \quad (D.5)$$

is the solution of

$$P = \Phi P, \quad (D.6)$$

where the elements of $[\Phi]$ are given by equation D.4. The elements of $P$ are the initial state probabilities.

Then the probability that the encoder is in state $\sigma_i$ is denoted by

$$P(u_i = \sigma_i) = p_i \forall i \quad (D.7)$$

and hence the state sequence is a stationary Markov chain. Use of equation D.7 in equation D.3 gives

$$P(\gamma_i = S_{ij}^k) = p_{ij}^k \forall i, \quad (D.8)$$

and hence $\gamma_i$ is stationary.

**D.3 The CPM system**

Here the formulas concerning the CPM system that are important in this appendix are repeated. For an explanation of the symbols and equations see section 2.4 of this report. The CPM signal is written as

$$S(t) = \sqrt{2E \over T_s} \cos\{2\pi f_c t + \phi(t) + \phi_0\}, \quad (D.9)$$

where

$$\phi(t) = \pi h \sum_{n=-\infty}^{\infty} \gamma_n g(t - nT_s), \quad (D.10)$$

is the phase deviation, and with phase response function
\[ g(t) = \int_{-\infty}^{t} f(\tau) d\tau. \]  

(D.11)

Important is that the function \( f(t) \), which is called the frequency pulse, is limited to the interval \( 0 \leq t < LT_s \) and that it integrates to 1. This has the important consequence that

\[ g(t) = \begin{cases} 
0, & t \leq 0 \\
1, & t \geq LT_s 
\end{cases} \]  

(D.12)

Another important signal in this report is the complex-valued signal \( V(t) \) defined by

\[ V(t) \triangleq \exp\{j(\phi(t) + \phi_0)\}, \]  

(D.13)

where \( \phi(t) \) is given in equation D.10.

### D.4 Relation between the PSD of the transmitted signal and the PSD of the complex-valued signal

The PSD of the complex-valued signal \( V(t) \) is used to calculate the PSD of the real valued CPM signal \( S(t) \). The PSDs are represented by \( P_v(f) \) and \( P_s(f) \) respectively. In this section the representation of \( P_s(f) \) in terms of \( P_v(f) \) is derived. It is not difficult to see that \( S(t) \) can be written in terms of \( V(t) \) as

\[
S(t) = \sqrt{\frac{2E}{T_s}} \Re \left\{ \exp\{j2\pi f_c t\} V(t) \right\} \\
= \sqrt{\frac{2E}{T_s}} \left[ \exp\{j2\pi f_c t\} V(t) + \exp\{-j2\pi f_c t\} V^*(t) \right].
\]  

(D.14)

To find the expression for the PSD of the real CPM signal, the autocorrelation function must be calculated. To distinguish between the autocorrelation function of \( S(t) \) and the autocorrelation function of \( V(t) \) the subscripts \( s \) and \( v \) are added. This gives

\[
R_s(t, \tau) = E[S(t + \tau)S(t)] \\
= \frac{E}{2T_s} \left\{ \exp\{j2\pi f_c (t + \tau)\} V(t + \tau) + \exp\{-j2\pi f_c (t + \tau)\} V^*(t + \tau) \right\} \\
\cdot \left\{ \exp\{j2\pi f_c t\} V(t) + \exp\{-j2\pi f_c t\} V^*(t) \right\} \\
= \frac{E}{2T_s} \left\{ \exp\{j2\pi f_c t\} V(t + \tau) V^*(t) + \exp\{-j2\pi f_c t\} V^*(t + \tau) V(t) \right\}
\]
+ \exp\{j2\pi f_c(2t + \tau)\} V(t + \tau)V(t) + \exp\{-j2\pi f_c(2t + \tau)\} V^*(t + \tau)V^*(t)\]

\[= \frac{E}{2T_s} \left( \exp\{j2\pi f_c\tau\} R_v(t, \tau) + \exp\{-j2\pi f_c\tau\} R^*_v(t, \tau) \right.
\[+ \exp\{j2\pi f_c(2t + \tau)\} R_{vv'}(t, \tau) \]
\[+ \left. \exp\{-j2\pi f_c(2t + \tau)\} R^*_{vv'}(t, \tau) \right) , \]  \hspace{1cm} (D.15)

where

\[R_{vv'}(t, \tau) = E[V(t + \tau)V(t)] \]  \hspace{1cm} (D.16)

is the crosscorrelation between \(V(t)\) and \(V^*(t)\). The average autocorrelation function \(R_\sigma(\tau)\) is

\[R_\sigma(\tau) = A[R_\sigma(t, \tau)] \]
\[= \frac{E}{2T_s} \left( \exp\{j2\pi f_c\tau\} R_v(\tau) + \exp\{-j2\pi f_c\tau\} R^*_v(\tau) \right.
\[+ \exp\{j2\pi f_c\tau\} A \left[ \exp\{j4\pi f_c t\} R_{vv'}(t, \tau) \right] \]
\[+ \left. \exp\{-j2\pi f_c\tau\} A \left[ \exp\{-j4\pi f_c t\} R^*_{vv'}(t, \tau) \right] \right) , \]  \hspace{1cm} (D.17)

where

\[R_v(\tau) = A[R_v(t, \tau)] , \]  \hspace{1cm} (D.18)

and

\[R^*_v(\tau) = A[R^*_v(t, \tau)] . \]  \hspace{1cm} (D.19)

Note that \(A[\cdot]\) denotes time average.

Now it is shown that, subject to a mild condition, the last two terms in equation D.17 do not contribute significantly to the PSD of \(S(t)\). For all schemes discussed in this report it may be safely assumed that \(P_v(f)\) is strictly band-limited, or at least negligibly small at frequencies \(|f| \geq f_c\);

\[P_v(f) = 0, \ |f| \geq f_c. \]  \hspace{1cm} (D.20)

Now define

\[W(t) \overset{\Delta}{=} \exp\{j2\pi f_c t\} V(t) . \]  \hspace{1cm} (D.21)
Then the autocorrelation function of \( W(t) \) is

\[
R_w(t, \tau) = E[W(t + \tau)W^*(t)]
= E[\exp\{j2\pi f_c(t + \tau)\}V(t + \tau)\exp\{-j2\pi f_c\tau\}V^*(t)]
= \exp\{j2\pi f_c\tau\}R_v(t, \tau),
\]

where \( R_v(t, \tau) \) is the autocorrelation function of \( V(t) \). The average autocorrelation function of \( W(t) \) is

\[
R_w(\tau) = \exp\{j2\pi f_c\tau\}R_v(\tau),
\]

where \( R_v(\tau) \) is the average autocorrelation function of \( V(t) \). The PSD of \( W(t) \) is the Fourier transform of \( R_w(\tau) \):

\[
P_w(f) = \int_{-\infty}^{\infty} R_w(\tau) \exp\{-j2\pi f \tau\} d\tau
= \int_{-\infty}^{\infty} R_v(\tau) \exp\{-j2\pi (f - f_c)\tau\} d\tau
= P_v(f - f_c).
\]

The Fourier transform of \( W^*(t) \) is

\[
P_{w^*}(f) = P_v(-f - f_c),
\]

where the last result is obtained by using the fact that \( P_v(f) \) is a real valued function (this will be proved in the last alinea of this section). Note that since it is assumed that \( P_v(f) \) is strictly band-limited (see D.20), the PSDs \( P_w(f) \) and \( P_{w^*}(f) \) do not overlap. Now the average crosscorrelation function of \( W(t) \) and \( W^*(t) \) is

\[
R_{ww^*}(\tau) = A[R_{ww^*}(t, \tau)]
= A[E[W(t + \tau)W(t)]]
= A[E[\exp\{j2\pi f_c(t + \tau)\}V(t + \tau)\exp\{j2\pi f_c\tau\}V(t)]]
= \exp\{j2\pi f_c\tau\}A[\exp\{j4\pi f_c\tau\}R_{vv^*}(t, \tau)],
\]

where \( R_{vv^*}(t, \tau) \) is the crosscorrelation function defined in equation D.16. The Fourier transform of equation D.28 is the corresponding cross PSD \( P_{ww^*}(f) \). It was already stated that the PSDs of \( P_w(f) \) and \( P_{w^*}(f) \) do not overlap. This means that the cross PSD \( P_{ww^*}(f) \) must be zero for all \( f \). The cross PSD \( P_{ww^*}(f) \) is zero if and only if the cross
The autocorrelation function method

In this section the autocorrelation function method of P.K.M. Ho and P.J. McLane [11] is explained. The autocorrelation function method was described before by many authors. The contribution of Ho and McLane is the elaboration of the method such that it can be used to calculate the PSD of CPM signals which are generated by a CPM modulator excited by correlated input symbols. Important is that the computational complexity of this method is linear in the pulse length \( L \). This is a remarkable result since at first it seems that the computational complexity is exponential in \( L \). In this section the manipulations performed by Ho and McLane are discussed and special attention is given to the computational complexity of the different formulas. In this section the subscript \( v \) in \( R_v(\tau) \) etc. is omitted.

In the main part of this report it was already shown that the PSD of the complex-valued signal \( V(t) \) is given by

\[
P_v(f) = \frac{2}{T_s} \text{Re} \left[ \int_{t=0}^{T_s} \int_{\tau=0}^{\infty} R(t, \tau) \exp\{-j2\pi f\tau\} d\tau dt \right],
\]

where

\[
R(t, \tau) = E[V(t + \tau)V^*(t)],
\]

and \( V(t) \) is given by equation D.13.

The autocorrelation function must be integrated over the infinite area given by \( 0 \leq t < T_s \) and \( \tau \geq 0 \). Ho and McLane derived a recursive procedure to determine \( R(t, \tau) \). The first step is to calculate \( R(t, \tau) \) in the area given by \( 0 \leq t < T_s, \tau \geq 0 \) and \( mT_s \leq t + \tau < (m+1)T_s \), where \( m \) is any non-negative integer. See figure D.2.

Use of equation D.10 in equation D.13 and application of the result in equation D.31 yields
Figure D.2: Area of integration of the autocorrelation function

\[ R(t, \tau) = E \left[ \exp \left\{ j(\pi h \sum_{n=-\infty}^{\infty} \gamma_n g(t + \tau - nT_s) + \phi_0) \right\} \right. \]
\[ \cdot \exp \left\{ -j(\pi h \sum_{n=-\infty}^{\infty} \gamma_n g(t - nT_s) + \phi_0) \right\} \]
\[ = E \left[ \exp \left\{ j\pi h \sum_{n=-\infty}^{\infty} \gamma_n (g(t + \tau - nT_s) - g(t - nT_s)) \right\} \right] \]
\[ = E \left[ \prod_{n=-\infty}^{\infty} \exp \left\{ j\pi h\gamma_n p(t - nT_s, \tau) \right\} \right]. \quad (D.32) \]

where

\[ p(t, \tau) = g(t + \tau) - g(t). \quad (D.33) \]

Because the phase pulse \( g(t) \) equals 1 for \( t \geq L T_s \) and 0 for \( t \leq 0 \), and because the point \((t, \tau)\) is in the bounded area mentioned above, the autocorrelation function of equation D.32 can be simplified. Now, it is shown that only for a limited number of values \( n \), \( p(t - nT_s, \tau) \neq 0 \). If \( p(t - nT_s, \tau) = 0 \) this results in a factor equal to 1 in equation D.32, which can be omitted. First, note that \( t + \tau - nT_s \geq t - nT_s \) since \( \tau \geq 0 \) (this is true for the whole area of integration). Thus, if \( t + \tau - nT_s \leq 0 \) then also \( t - nT_s \leq 0 \) and thus \( p(t - nT_s, \tau) = 0 \), which means that that \( n \) can be omitted in the product term. The largest value for \( t + \tau \) in the bounded area is \((m + 1)T_s\). Thus \( n \) can be omitted.
if \((m + 1)T_s - nT_s \leq 0\), or \(n \geq m + 1\). On the other hand, if \(t - nT_s \geq LT_s\) then \(t + \tau - nT_s \geq LT_s\) and again \(p(t - nT_s, \tau) = 0\) which means that this \(n\) may be omitted also. The smallest value for \(t\) in the bounded area is 0. Thus \(n\) is omitted if \(-nT_s \geq LT_s\), or \(n \leq -L\). Equation D.32 can be written as

\[
R(t, \tau) = E \left[ \prod_{n=-L+1}^{m} \exp\{j\pi h \gamma_n p(t - nT_s, \tau)\} \right],
\]

\(0 \leq t < T_s, \ mT_s \leq t + \tau < (m + 1)T_s, \ \tau \geq 0\). (D.34)

If the transmitted symbols in the sequence \(\{\gamma_{-L+1}, \gamma_{-L+2}, \ldots, \gamma_{m}\}\) could all be chosen independently then the expectation would be over \(M^{L+m}\) sequences, where \(M\) is the number of channel symbols. As explained in the discussion of the Markov chain approach (see section 3.3), the number of possible sequences is smaller. This means that the expectation is over \(N \cdot K^{L+m}\) sequences, where \(K\) is the total number of one step transitions from each state, and \(N\) is the number of states. Note that the number of sequences is exponential in \(m\) and \(L\).

Now a recursive relation is derived for the calculation of the autocorrelation function. First, assume that the first symbol that contributes to the result in equation D.34, is equal to \(s_{m'}\). Thus \(\gamma_{-L+1} = s_{m'}\). This symbol conditional autocorrelation function is given by

\[
R_m(t, \tau \mid \gamma_{-L+1} = s_{m'}) = \exp\{j\pi h s_m p(t + (L - 1)T_s, \tau)\}
\]

\[\cdot E \left[ \prod_{n=-L+2}^{m} \exp\{j\pi h \gamma_n p(t - nT_s, \tau)\} \mid \gamma_{-L+1} = s_{m'} \right] \text{D.35}\]

where the subscript \(m\) denotes that the calculations are valid for points \((t, \tau)\) in the bounded area determined by \(m\). Note that the expectation in equation D.35 is over \(K^{L+m-1}\) sequences. The second step made by Ho and McLane was to introduce another conditional autocorrelation function \(\psi_m(t, \tau)\). This function represents the autocorrelation of \(V(t)\), assuming that the first state that contributes to the result in equation D.34 is equal to \(\sigma_i\). Thus \(\gamma_{-L+1} = \sigma_i\). This function is called the state conditional autocorrelation function. Expression of the state conditional autocorrelation function in symbol conditional autocorrelation functions of equation D.35 gives

\[
\psi_m(t, \tau) = \sum_{j} \sum_{k=1}^{N_p} t_{ij} R_m(t, \tau \mid \gamma_{-L+1} = s_{m'}), \text{D.36}\]

where the first summation is over those states \(\sigma_j\) that can be reached from \(\sigma_i\) in one step. Note that \(K^{L+m}\) sequences are involved in equation D.36. The autocorrelation function for points \((t, \tau)\) in the bounded area is now expressed as...
\[ R(t, \tau) = \sum_{i=1}^{N} p_i \Psi_m^i(t, \tau) = \mathcal{P}(\Psi_m(t, \tau)), \]

\[ 0 \leq t < T_s, \ mT_s \leq t + \tau < (m + 1)T_s, \ \tau \geq 0, \]  

(D.37)

where

\[ \Psi_m(t, \tau) = [\psi_m^1(t, \tau), \psi_m^2(t, \tau), \ldots, \psi_m^N(t, \tau)] \]  

(D.38)

and \( p = [p_1, p_2, \ldots, p_N] \). The elements \( p_i \) denote the probability that the encoder is in state \( \sigma_i \) (as explained in the second section of this appendix). Note that the calculation of \( \Psi_m(t, \tau) \) involves evaluations for \( NK^{L+m} \) sequences.

Now, let \( m \) in equation D.35 be increased by one. For this case, taking into account that the pulse length is \( LT_s \), the symbol conditional autocorrelation function is

\[ R_{m+1}(t, \tau | \gamma_{-L+1} = S_{ij}^k) = \exp\{j\pi h S_{ij}^k p(t + (L - 1)T_s, \tau)\} \]

\[ \cdot \mathbb{E}\left[ \prod_{n=-L+2}^{n+1} \exp\{j\pi h \gamma_{n} p(t - nT_s, \tau)\} | \gamma_{-L+1} = S_{ij}^k \right]. \]  

(D.39)

Given \( \gamma_{-L+1} = S_{ij}^k \), for certain \( i, j \) and \( k, \gamma_{-L+2} \) must be \( S_{j_1 k_1}^{k_1} \), for certain \( j_1 \) and \( k_1 \). Thus the average of the second term in equation D.39 is

\[ \mathbb{E}\left[ \left( \prod_{n=-L+2}^{n+1} \exp\{j\pi h \gamma_{n} p(t - nT_s, \tau)\} \right) | \gamma_{-L+1} = S_{ij}^k \right] \]

\[ = \sum_{j_1} \sum_{k_1=1}^{N_p} \left\{ \left( \prod_{n=-L+3}^{n+1} \exp\{j\pi h S_{j_1 k_1}^{k_1} p(t + (L - 2)T_s, \tau)\} \right) \right. \]

\[ \cdot \mathbb{E}\left[ \prod_{n=-L+3}^{n+1} \exp\{j\pi h \gamma_{n} p(t - nT_s, \tau)\} | \gamma_{-L+2} = S_{j_1 k_1}^{k_1} \right]\left\}, \]  

(D.40)

where the first summation is over those states \( \sigma_{j_1} \) that are reachable from \( \sigma_j \) in one step. Equation D.40 is the same equation than the one obtained when equation D.35 is used in equation D.36. The only differences are that the calculations in equation D.40 are performed one time slot of length \( T_s \) later in time and the first state concerning the calculations is assumed to be \( \sigma_j \) instead of state \( \sigma_i \). Since the signal \( V(t, \tau) \) is cyclostationary with period \( T_s \) (which was explained before) the RHS of equation D.40 is
actually $\psi_m^j(t - T, \tau)$. Thus the conditional correlation function according to equation D.39 is

$$R_{m+1}(t, \tau \mid \gamma_{-L+1} = S^k_{ij}) = \exp\{j\pi h S^k_{ij} p(t + (L - 1)T, \tau)\} \psi_m^j(t - T, \tau). \quad (D.41)$$

Replacing $m$ by $m+1$ in equation D.36 and substituting equation D.41 into equation D.36 yields an expression for the state conditional autocorrelation function in the next slot:

$$\psi_{m+1}^j(t, \tau) = \sum_j \left\{ \sum_{k=1}^{N_p} t_{ij}^k \exp\{j\pi h S^k_{ij} p(t + (L - 1)T, \tau)\} \right\} \psi_m^j(t - T, \tau), \quad (D.42)$$

which is a recursive equation for $\psi_m^j(t, \tau)$. In the discussion after equation D.36, it was stated that $K^{L+m}$ sequences ($\gamma_{-L+1}, \gamma_{-L+2}, \ldots, \gamma_m$), have to be evaluated for the calculation of $\psi_m^j(t, \tau)$, using the formulas in equations D.35 and D.36. According to equation D.37 the evaluation is needed for all $N$ starting states $\sigma_i$, or, in other words, $\Psi_m(t, \tau)$ has to be calculated. For $\Psi_m(t, \tau)$ a total of $NK^{L+m}$ sequences has to be evaluated. Using the same formulas for the calculation of all $\psi_{m+1}^j(t, \tau)$ in $\Psi_{m+1}(t, \tau)$] requires the evaluation of $NK^{L+m+1}$ sequences. Now examine equation D.42. It is obvious that eventually $\psi_m^j(t - T, \tau)$ has to be evaluated for all states $\sigma_i$. This means the evaluation of $NK^{L+m}$ sequences. If the first summation in equation D.42 would go over all states in stead of those that are reachable from state $\sigma_i$ in one step, then the calculation of the $N$ elements of $\Psi_{m+1}(t, \tau)$] needs an additional set of calculations that is proportional to $N^2$ ($N$ times a summation of $N$ terms). This is where the calculations become linear in the pulse length $L$.

To obtain a more compact notation, Ho and McLane introduced the function

$$q_{ij}^k(t) = \exp\{j\pi h S^k_{ij} g(t)\}. \quad (D.43)$$

It is not difficult to see, that since $g(t) = 0$ for $t \leq 0$ and 1 for $t \geq LT,\,$

$$q_{ij}^k(t) = \begin{cases} 1, & t \leq 0 \\ \exp\{j\pi h S^k_{ij}\}, & t \geq LT \end{cases}. \quad (D.44)$$

Combining equations D.33 with equation D.43 implies that $\exp\{j\pi h S^k_{ij} p(t, \tau)\}$ is equal to $q_{ij}^k(t + \tau) : q_{ij}^k(t)^*$. Thus, by letting

$$x_{ij}(t, \tau) = \sum_{k=1}^{N_p} t_{ij}^k q_{ij}^k(t + \tau) q_{ij}^k(t)^* \quad (D.45)$$

equation D.42 can be rewritten in vector notation as

© Philips Electronics N.V. 75
where the \((i,j)\)th element of \([X(t,\tau)]\) is given in equation \(D.45\). In equation \(D.46\) the change of the complexity to be linear in \(L\) is more visible. According to the discussion after equation \(D.38\), the calculation of \(\Psi_m(t,\tau)\) requires the evaluation of \(NK^{L+m}\) sequences, thus the evaluation of \(\Psi_{m+1}(t,\tau)\) requires \(NK^{L+m+1}\) sequences to be evaluated. Examining the RHS of equation \(D.46\) shows that \(\Psi_{m+1}(t,\tau)\) can be calculated bij first calculating \(\Psi_m(t,\tau)\), which means the evaluation of \(NK^{L+m}\) sequences. Then a matrix-vector multiplication has to be performed. Since the matrix and the vector are of size \(N\), this requires an \textbf{additional} amount of calculations proportional to \(N^2\). Equation \(D.46\) is a recursive formula for the calculation of \(\Psi_m(t,\tau)\). Repeated use of equation \(D.46\) gives

\[
\Psi_m(t,\tau) = \left( \prod_{n=1-L}^{m-L} [X(t-nT_\tau,\tau)] \right) \Psi_0(t-mT_\tau,\tau). \tag{D.47}
\]

As a result of the assumption that the point \((t,\tau)\) is in the bounded area, the expression in the large parentheses in equation \(D.47\) can be simplified. It is shown that the argument of the first \(q_{ij}\) function which appears on the RHS of equation \(D.45\) is always larger than or equal to \(LT_\sigma\) when used in equation \(D.47\), which means that it can be replaced by by the constant \(q_{ij}^k(LT_\sigma)\). Use of equation \(D.45\) in equation \(D.47\) shows that the parameter of this first \(q_{ij}\) function is \(t-nT_\sigma+\tau\). In the bounded area, the smallest value for \(t+\tau\) is \(mT_\sigma\). The largest value for \(nT_\sigma\) is \((m-L)T_\sigma\). Thus \(t-nT_\sigma+\tau \geq LT_\sigma\), which is what had to be shown. As a result the matrices \(X\) in equation \(D.47\) are independent of \(\tau\). To emphasize this, the matrix \([Z(t)]\) is introduced of which the \((i,j)\)th element is defined by

\[
z_{ij}(t) = \sum_{k=1}^{N_p} t_{ij}^k q_{ij}^k (LT_\sigma) q_{ij}(t)^*. \tag{D.48}
\]

Using this matrix, equation \(D.47\) becomes

\[
\Psi_m(t,\tau) = \left( \prod_{n=1-L}^{m-L} [Z(t-nT_\sigma)] \right) \Psi_0(t-mT_\sigma,\tau)
\]

\[0 \leq t < T_\sigma, \quad mT_\sigma \leq t+\tau < (m+1)T_\sigma, \quad \tau \geq 0. \tag{D.49}
\]

The last step in finding the representation for \(R(t,\tau)\) is to determine \(\Psi_0(t,\tau)\), the initial value of \(\Psi_m(t,\tau)\). Then, for any point \((t,\tau)\) in the area of integration, the autocorrelation function can be calculated. If \(m = 0\) in equation \(D.35\) then the expectation on the RHS is obtained by averaging over \(K^{L-1}\) sequences \((\gamma_{-L+2}, \gamma_{-L+3}, \ldots, \gamma_0)\). Combining equations \(D.35\) and \(D.36\), and setting \(m = 0\), shows that the calculation of the elements \(\psi_0(t,\tau)\) of \(\Psi_0(t,\tau)\) are found by the averaging of all \(K^L\) sequences \((\gamma_{-L+1}, \gamma_{-L+2}, \ldots, \gamma_0)\). At
first sight, the computational complexity seems to be exponential in the baseband pulse length $L$. However, Ho and McLane [11] showed that $\Psi_0(t, \tau)$ can be calculated with a complexity linear in $L$.

The output sequences $(\gamma_{-L+1}, \gamma_{-L+2}, \ldots, \gamma_0)$ can be written in the form

$$S_{i_{-L+1}, j_{-L+2}, \ldots, i_{-L+2}, j_{-L+3}, \ldots, i_0, j_1}.$$  

(D.50)

Given that $\gamma_{-L+1} = S_{ij}$, each of the above sequences occurs with probability

$$\prod_{n=-L+2}^{0} t_{i_n, j_{n+1}}^{k_{n+1}}.$$  

(D.51)

Putting $m = 0$ together with the above information into equation D.35 and then combining equations D.35 and D.36 yields

$$\psi_0(t, \tau) = \left( \sum_{l_{-L+2}} \sum_{l_{-L+3}} \cdots \sum_{l_1} \right) \left( \sum_{k_{-L+2}=1} \sum_{k_{-L+1}=1} \cdots \sum_{k_1=1} \right) \prod_{n=-L+1}^{0} x_{i_n, j_{n+1}}^{k_{n+1}} (t - nT_s, \tau).$$  

(D.52)

where

$$x_{ij}^k(t, \tau) = t_{ij}^k q_{ij}^k(t + \tau) q_{ij}^k(t)^*.$$  

(D.53)

Interchanging the second group of summations with the product term yields

$$\psi_0(t, \tau) = \left( \sum_{l_{-L+2}} \sum_{l_{-L+3}} \cdots \sum_{l_1} \right) \prod_{n=-L+1}^{0} x_{i_n, j_{n+1}}(t - nT_s, \tau),$$  

(D.54)

where the first state $\sigma_{-L+1} = \sigma_i$ and $x_{ij}(t, \tau)$ is defined in equation D.45. Equation D.54 can be written in matrix form as

$$\Psi_0(t, \tau) = \left( \prod_{n=-L+1}^{0} [X(t - nT_s, \tau)] \right) \mathbf{1},$$  

(D.55)

where

© Philips Electronics N.V. 77
\[ \mathbf{1} = [1,1,\ldots,1] \tag{D.56} \]

and the \((i,j)\)th element of \([X(t,\tau)]\) is defined in equation D.45.

The recursive procedure to calculate \(R(t,\tau)\) is represented by the equations D.37, D.49 and D.55. Where the elements of \(Z(t,\tau)\) are defined in equation D.48 and the elements of \(X(t,\tau)\) are defined in equation D.45. The last two equations use the function \(q(t)\) which is defined in equation D.43.

The complexity of the computation of the autocorrelation function using the equations derived so far is linear in \(m\). Since larger values of \(\tau\) imply larger values of \(m\), the complexity of the calculation of \(R(t,\tau)\) for those values of \(\tau\), grows without bounds. Fortunately for \(t+\tau \geq LT_s\) the expression for \(R(t,\tau)\) can be further simplified. Note that the statement \(t+\tau \geq LT_s\) is equal to the statement \(m \geq L\). In this region, the double integral of equation D.30 can be replaced by an infinite matrix series. This is shown next.

Consider the matrix \([Z(t-nT_s)]\). The \((i,j)\)th element of \([Z(t)]\) is defined in equation D.48. Since \(0 \leq t < T_s\), \(t-nT_s < 0\), for \(n > 0\). If the argument of \(z_{ij}\) in equation D.48 is negative, then, using equation D.44 \(z_{ij}(t-nT_s)\) can be simplified to the constant

\[ w_{ij} = z_{ij}(t-nT_s) = \sum_{k=1}^{N_p} t_{ij}^k q_{ij}(LT_s) \quad n > 0, \tag{D.57} \]

where \(w_{ij}\) is the \((i,j)\)th element of the matrix \([W]\). Thus, for \(m \geq L\), the product term in equation D.49 can be broken down into the product of two terms:

\[ \prod_{n=1-L}^{m-L} [Z(t-nT_s)] = [H(t)][W]^{m-L} \quad m \geq L, \tag{D.58} \]

where

\[ [H(t)] = \prod_{n=1-L}^{0} [Z(t-nT_s)]. \tag{D.59} \]

Note that, by definition, \([W]^0\) is the identity matrix \([I]\).

Next, consider the term \(\Psi_0(t-mT_s,\tau)\), which appears in equation D.49. Using equation D.55, \(\Psi_0(t-mT_s,\tau)\) can be written as

\[ \Psi_0(t-mT_s,\tau) = \left( \prod_{n=-L+1}^{0} [X(t-mT_s-nT_s,\tau)] \right) 1 \]

© Philips Electronics N.V.
where the \((i,j)\)th element of \([X(t, \tau)]\) is defined in equation D.45. Since it is assumed that 
\(m \geq L\), the smallest value of \(n'\) is 1. And since \(0 \leq t < T_s\), \(t - n'T_s\) in equation D.60 is 
always negative. Using this fact in combination with equations D.43 and D.44 gives

\[
x_{ij}(t - nT_s, \tau) = \sum_{k=1}^{N_p} t_{ij}^k q_{ij}^k(t - nT_s + \tau)q_{ij}^k(t - nT_s)\]

\[
= \sum_{k=1}^{N_p} t_{ij}^k q_{ij}^k(t + \tau - nT_s)
\]

\[
= y_{ij}(t + \tau - nT_s),
\]

\(n > 0,\) \hspace{1cm} (D.61)

where \(y_{ij}(\tau)\) is the \((i,j)\)th element of \([Y(\tau)]\). Thus, if \(m \geq L\), from equation D.61 it follows that

\[
\Psi_0(t - mT_s, \tau) = A(t + \tau - mT_s) \quad m \geq L,
\]

\(\) \hspace{1cm} (D.62)

where

\[
A(\tau) = \left( \prod_{n=1-L}^{0} [Y(\tau - nT_s)] \right)^{-1}.
\]

\(\) \hspace{1cm} (D.63)

Finally, combining equations D.37, D.49, D.58 and D.62 yields

\[
R(t, \tau) = E[H(t)][W]^{m-L}A(t + \tau - mT_s),
\]

\(m \geq L, \quad 0 \leq T_s \quad mT_s \leq t + \tau < (m + 1)T_s.\) \hspace{1cm} (D.64)

Equation D.64 is the expression for \(R(t, \tau)\) for \(t + \tau \geq LT_s\). For \(t + \tau < LT_s\), \(R(t, \tau)\) is 
given by equations D.37, D.49 and D.55. Thus, from equation D.30, and using these two 
forms of \(R(t, \tau)\), \(P_v(f)\) can be written as

\[
P_v(f) = 2\text{Re}[P_1(f) + P_2(f)],
\]

\(\) \hspace{1cm} (D.65)

where, \(P_1(f)\) represents the integration for \(t + \tau < LT_s\), and is defined as
\[ P_1(f) = \frac{1}{T_s} \int_{t=0}^{T_s} \int_{\tau=0}^{LT_s-t} R(t, \tau) \exp\{-j2\pi f \tau\} d\tau dt, \quad (D.66) \]

and, \( P_2(f) \) represents the integration for \( t + \tau \geq LT_s \), and is defined as

\[ P_2(f) = \frac{1}{T_s} \int_{t=0}^{T_s} \int_{\tau=LT_s-t}^{\infty} R(t, \tau) \exp\{-j2\pi f \tau\} d\tau dt \quad (D.67) \]

Changing the order of integration in equation D.66 gives

\[ P_1(f) = \frac{1}{T_s} \int_{\tau=0}^{(L-1)T_s} \left( \int_{t=0}^{T_s} R(t, \tau) dt \right) \exp\{-j2\pi f \tau\} d\tau \\
+ \frac{1}{T_s} \int_{\tau=(L-1)T_s}^{LT_s} \left( \int_{t=0}^{LT_s-\tau} R(t, \tau) dt \right) \exp\{-j2\pi f \tau\} d\tau \\
= \frac{1}{T_s} \int_{\tau=0}^{LT_s} R_1(\tau) \exp\{-j2\pi f \tau\} d\tau, \quad (D.68) \]

where

\[ R_1(\tau) = \begin{cases} 
\int_{t=0}^{T_s} R(t, \tau) dt & 0 \leq \tau < (L-1)T_s \\
\int_{t=0}^{LT_s-\tau} R(t, \tau) dt & (L-1)T_s \leq \tau < LT_s.
\end{cases} \quad (D.69) \]

The advantage of changing the order of integration is that the integration in the \( t \)-direction only has to be performed once to obtain \( R_1(\tau) \). Subsequently to calculate \( P_1(f) \) for different values of \( f \) only a line integral has to be evaluated instead of a surface integral.

Use of equation D.64 into D.67 gives

\[ P_2(f) = \frac{1}{T_s} \int_{t=0}^{T_s} \sum_{m=L}^{\infty} \int_{\tau=mT_s-t}^{(m+1)T_s-t} E[H(t)] [W]^{m-L} A(t + \tau - mT_s) \\
\exp\{-j2\pi f \tau\} d\tau dt. \quad (D.70) \]

Use of \( \tau' = t + \tau - mT_s \) gives

\[ P_2(f) = \frac{1}{T_s} \int_{t=0}^{T_s} \sum_{m=L}^{\infty} \int_{\tau'=t+mT_s}^{T_s} E[H(t)] [W]^{m-L} A(\tau') \\
\exp\{-j2\pi f (\tau' - t + mT_s)\} d\tau' dt. \quad (D.71) \]

Omitting the accent and further manipulation of the formula yields
\[ P_2(f) = \frac{1}{T_s} \int_{t=0}^{T_s} \mathcal{P}[H(t)] \sum_{m=L}^{\infty} [W]^{m-L} \int_{\tau=0}^{T_s} A(\tau) \exp\{-j2\pi f \tau\} d\tau \]
\[ \times \exp\{-j2\pi f(-t + mT_s)\} dt \]
\[ = \frac{1}{T_s} \int_{t=0}^{T_s} \mathcal{P}[H(t)] \sum_{m=L}^{\infty} [W]^{m-L} A(f) \exp\{j2\pi ft\} \exp\{-j2\pi fmT_s\} dt \]
\[ = \frac{1}{T_s} \mathcal{P}[H(f)] \sum_{m=L}^{\infty} [W]^{m-L} \exp\{-j2\pi fmT_s\} A(f). \tag{D.72} \]

Letting \( m' = m - L \), \( P_2(f) \) can be expressed as

\[ P_2(f) = \frac{1}{T_s} \mathcal{P}[H(f)] \sum_{m'=0}^{\infty} [W]^{m'} \exp\{-j2\pi f(m' + L)T_s\} A(f) \]
\[ = \frac{1}{T_s} \exp\{-j2\pi fLT_s\} \mathcal{P}[H(f)] \]
\[ \times \left( \sum_{n=0}^{\infty} [W]^{n} \exp\{-j2\pi fnT_s\} \right) A(f). \tag{D.73} \]

In these equations

\[ A(f) = \int_{0}^{T_s} A(\tau) \exp\{-j2\pi f \tau\} d\tau \tag{D.74} \]

and

\[ [H(f)] = \int_{0}^{T_s} \exp\{+j2\pi ft\} dt. \tag{D.75} \]

Note that \( P_1(f) \) is the integral of a finite function and its contribution to \( P_2(f) \) is therefore finite. The infinite matrix series in equation D.73 may result in an infinite contribution of \( P_2(f) \) to \( P_3(f) \). If the contribution is infinite this means that the PSD of \( S(t) \) contains line components. Ho and McLane stated that line components did not occur in the PSDs of the schemes they investigated. In the PSDs calculated for the schemes discussed in this report, no line components were found either. For the discussion of line components see Ho and McLane [11].

If no line components are present the infinite matrix series in equation D.73 converges. In this case Ho and McLane showed that the equation D.73 can be rewritten as
\[ P_2(f) = \frac{1}{T_s} \exp\{-j2\pi f LT_s\} P[H(f)] \times \{[I] - \exp\{-j2\pi f T_s\}[W]\}^{-1} A(f), \] (D.76)

where \([\cdot]^{-1}\) denotes inverse.
Appendix E

Eye-patterns and PSDs
E.1 Eye-patterns (1,3) RLL GMSK
E.2 Eye-patterns (1,7) RLL GMSK

© Philips Electronics N.V.
E.3 Eye-patterns (2,7) RLL GMSK
E.4 Eye-patterns uncoded GMSK
E.5 Eye-patterns 2B1Q GMSK
E.6 PSDs (1:3) RIT GMSK
- y1-axis -
P.f1 0.2
P.f2 0.25
P.f3 0.50
P.f4 1.0
\( \beta \tau = 2 \beta \Delta \)
PSDs uncoded GMSK

\[ \beta_b^T = \beta_P \]

- y1-axis -
\[ p_{a1} \quad 0.2 \]
\[ p_{a2} \quad 0.25 \]
\[ p_{a3} \quad 0.5 \]
\[ p_{a4} \quad \infty \]

\( f \)